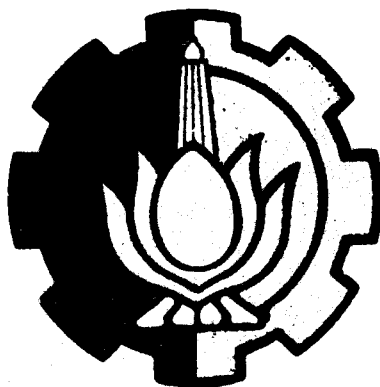


6105/ITS/H/94 ✓

**PERENCANAAN DAN PEMBUATAN  
SYNCHRONISASI GENERATOR TIGA PHASA  
YANG DIINTERFACEKAN KE IBM PC XT**

PERPUSTAKAAN ITS	
Tgl. Terima	20 OCT 1993
Terima Dari	H.
No. Agenda Prp.	1373 /TA



RSE  
621.39 6  
Put  
P-1  
1993

**OLEH :**

*Eko Hamijaya Putra*

**NRP : 2852200407**

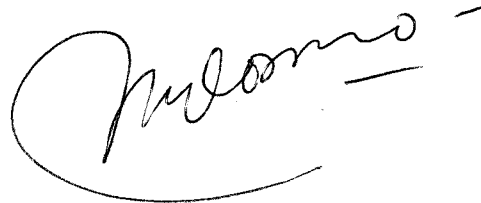
**JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
1993**

**PERENCANAAN DAN PEMBUATAN  
SYNCHRONISASI GENERATOR TIGA PHASA  
YANG DIINTERFACEKAN KE IBM PC XT**

**TUGAS AKHIR**

**Diajukan guna memenuhi sebagian  
persyaratan untuk memperoleh  
gelar Sarjana Teknik Elektro  
pada  
Bidang Studi Elektronika  
Jurusan Teknik Elektro  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya**

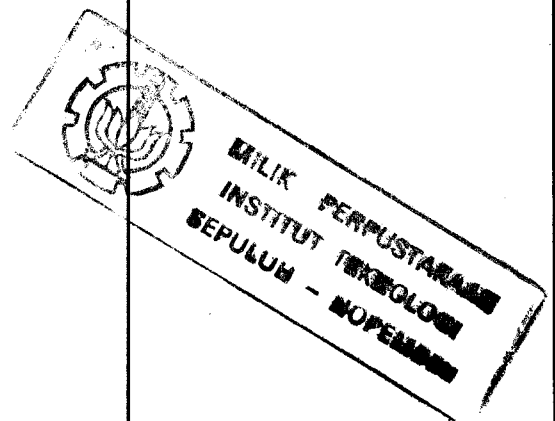
**Mengetahui / Menyetujui  
Dosen Pembimbing**



**Ir. MURDI ASMOROADJI**

**S U R A B A Y A**

**Agustus 1993**



## ABSTRAK

Energi listrik adalah energi yang paling banyak memegang peranan dalam menjalankan proses industri. Gangguan yang terjadi pada penyedia energi tersebut dalam hal ini generator akan mempengaruhi proses yang ada dalam proses produksi yang sedang berjalan. Untuk mengatasi hal tersebut dibutuhkan penyedia energi cadangan jadi bila ada masalah sewaktu - waktu mengganggu generator utama, atau terjadi kelebihan beban.

Generator dalam hal ini yang berfungsi sebagai penyedia energi mempunyai syarat - syarat tertentu untuk melakukan synchronisasi dengan generator yang lain. Dimana untuk melakukan synchronisasi tersebut syarat - syarat tersebut harus di penuhi.

Dari syarat - syarat tersebut yang merupakan input analog diolah dan dibandingkan antara dua generator yang kemudian dijadikan masukan bagi IBM PC yang berbasis mikroprosesor 8088 untuk melakukan synchronisasi.

Alat ini di desain untuk memonitor syarat - syarat kerja paralel tersebut dan memberikan informasi kepada IBM PC saat yang tepat untuk melakukan proses synchronisasi dan mengatur peralatan penggerak bila terjadi ketidak pemenuhan syarat - syarat kerja paralel tersebut.

## KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa atas segala karuniaNya dengan terselesaikanya Tugas Akhir yang berjudul :

PERENCANAAN DAN PEMBUATAN SINKRONISASI GENERATOR TIGA PHASA YANG DIINTERFACEKAN KE IBM PC - XT.

Tugas akhir ini adalah merupakan salah satu syarat yang harus ditempuh oleh setiap mahasiswa untuk meraih gelar kesarjanaan di bidang studi Elektronika jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember Surabaya.

Dalam pengerjaan Tugas Akhir ini, penulis melakukan penyusunan berdasarkan pada teori - teori yang telah didapat selama perkuliahan, literatur, bimbingan dari dosen wali serta pihak - pihak lain yang telah membantu dan mendorong semangat penyusunan. Oleh karena itu penulis pada kesempatan ini menyampaikan ucapan terima kasih yang sebesar - besarnya kepada :

- Ir. Murdi Asmoroadji, selaku dosen pembimbing dan dosen wali.
- Ir. Soetikno, selaku Koordinator Bidang Studi Elektronika, Jurusan teknik Elektro ITS.
- Ir. K. Astrowulan MSEE, selaku Ketua Jurusan Teknik Elektro ITS.
- Seluruh staf pengajar dan administrasi Jurusan Teknik Elektro ITS.

- Seluruh rekan - rekan mahasiswa jurusan Teknik elektro bidang studi Elektronika dan semua pihak yang telah memberikan bantuan, baik secara langsung maupun tidak langsung.

Akhirnya penulis berharap semoga segala sesuatu yang telah dihasilkan dalam pelaksanaan tugas akhir ini dapat bermanfaat bagi kemajuan Ilmu dan kesejahteraan umat manusia.

Surabaya, Juli 1993

Penulis

# DAFTAR ISI

BAB	HALAMAN
HALAMAN JUDUL .....	1
HALAMAN PENGESAHAN .....	11
HALAMAN PERSEMBAHAN .....	111
ABSTRAK .....	1v
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	x1
DAFTAR TABEL.....	xiii
 BAB I PENDAHULUAN	
I.1 Latar belakang.....	1
I.2 Permasalahan.....	2
I.3 Pembatasan masalah.....	3
I.4 Sistematika Pembahasan.....	3
I.5 Metodologi.....	4
 BAB II TEORI PENUNJANG	
II.1 Sistem pembangkit listrik .....	5
II.1.1 Generator .....	5
II.1.2 Pembangkitan arus dan tegangan.....	8
II.1.4 Kecepatan dan frekwensi.....	9
II.1.5 Perbedaan Phasa .....	11
II.2 Penggerak Utama .....	12
II.2.1 Motor sebagai Unit Penggerak .....	12
II.3 Kerja Paralel Generator .....	12

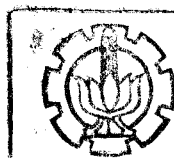
II.4	Arsitektur Mikrokomputer IBM PC .....	15
II.5	Mikroprosessor 8088 .....	16
II.5.1	Fungsi Kaki pada IC 8088.....	17
II.5.2	Pengadressan pada memory.....	18
II.5.3	Data Register.....	20
II.5.4	Pointer dan Indeks register.....	20
II.5.5	Segmen Register.....	20
II.5.6	Instruction Pointer register.....	21
II.6	Rangkaian Clock .....	22
II.7	Sistem board bus .....	22
II.8	Sistem memory .....	22
II.9	Sistem board timer counter.....	23
II.11	Sistem board DMA.....	24
II.12	Sistem board integrated I/O adapter....	25
II.13	Programable Peripheral Interface.....	26
II.14	Fungsi pin - pin 8255.....	27
II.14.1	Menghubungkan 8255 dengan 8088.....	29
II.15	Hubungan 8088 dengan slot yang dipakai untuk I/O slot .....	33
II.16	Siklus baca I/O port.....	38
II.17	Siklus Tulis I/O port .....	40
II.18	Buffer (penyangga).....	42
II.19	Pengubah Analog ke Digital .....	43
II.19.1	Paralel AD Converter.....	43
II.19.2	Dual Slope AD Converter.....	44
II.19.3	Successive Aproximation A/D Converter..	44
II.19.4	ADC 8088.....	46

II.19.4.1	Blok Diagram ADC 8088.....	46
BAB III PERENCANAAN		
III.1	Pendahuluan .....	51
III.2.	Blok Diagram.....	52
III.3.	Rangkaian Interface.....	52
III.4.	I/O Port mapping dan decoding .....	54
III.6.	Rangkaian Sensor Tegangan.....	58
III.7.	Rangkaian Sensor Frekwensi.....	59
III.7.1.	Rangkaian Buffer.....	59
III.7.2	Rangkaian Pengubah Sinus ke persegi....	59
III.8.	Rangkaian ADC.....	64
III.9.	Rangkaian Detektor Phasa.....	65
III.10.	Rangkaian Driver Motor Stepper .....	67
III.11.	Rangkaian Pengatur relay .....	68
III.12.	Perencanaan Perangkat lunak .....	66
III.12.1	Pendahuluan .....	68
III.12.2.	Menu Utama .....	69
III.12.3.	Menu syncro otomatis.....	70
III.10.4.	Menu Pengatur Putaran motor.....	71
BAB IV	PENGUKURAN DAN PENGUJIAN .....	78
IV.1	Pengujian .....	78
BAB V	KESIMPULAN .....	80
DAFTAR	PUSTAKA .....	81
LAMPIRAN :		
LISTING	PROGRAM .....	
RANGKAIAN	LENGKAP .....	
USULAN	TUGAS AKHIR .....	



## DAFTAR GAMBAR

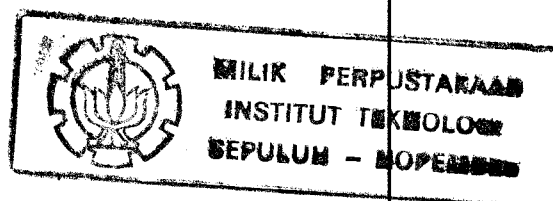
GAMBAR	HAL
2.1 Bagan sistem pembangkit listrik .....	5
2.2 Gambar Konstruksi Generator.....	6
2.3 a) Proses Pembangkitan listrik .....	7
2.3 b) Proses Pembangkitan listrik .....	7
2.4 Bentuk Gelombang yang dibangkitkan .....	8
2.5 Medan magnet rotor yang berputar .....	9
2.6 Kurva tegangan dan waktu .....	10
2.7 Kurva fasa "lead dan "lag".....	10
2.8 Hubungan Paralel antar generator .....	13
2.9 Hubungan busbar .....	13
2.10 Diagram phasor antar fasa .....	14
2.11 Diagram IBM PC XT .....	16
2.12 Pembangkitan Address memory pada 8088 .....	17
2.13 Register pada 8088 .....	19
2.14 Blok Diagram PPI 8255 .....	27
2.15 Format mode operasi 8255 .....	32
2.16 Slot IBM PC.....	34
2.17 Siklus baca I/O Port .....	39
2.18 Siklus tulis I/O port .....	41
2.19 Blok Diagram SA - ADC .....	45
3.1. Blok diagram peralatan yang direncanakan...	51
3.2. Penggunaan alamat I/O .....	54
3.3. Penggunaan alamat I/O pada slot IBM .....	55



3.4.	Rangkaian Decoder PPI satu .....	58
3.5.	Rangkaian decoder PPI dua .....	58
3.6.	Rangkaian sensor tegangan .....	59
3.7.	Rangkaian Penyangga .....	60
3.8.	Rangkaian Pengubah Gelombang Sinus ke Persegi	61
3.9.	Grafik yang menentukan waktu keluaran .....	62
3.10.	Rangkaian lengkap sensor frekwensi.....	63
3.11.	Rangkaian ADC.....	64
3.12.	Rangkaian Detektor Fasa.....	66
3.13.	Rangkaian Driver motor Stepper.....	67
3.14.	Rangkaian Pengatur relay .....	69
3.15.	Diagram alir menu Utama .....	70
3.16.	Diagram alir Sincronisdasi Otomatis .....	71
3.17.	Diagram Alir Deteksi Tegangan dan Frekwensi	72
4.1.	Foto Alat yang dibuat .....	79
4.2.	Foto Alat yang dibuat .....	79

## DAFTAR TABEL

TABEL	HAL
2.1. Pembagian lokasi memory .....	22
2.2. Pembagian memory address .....	23
2.3. Pembagian memory ROM .....	23
2.4. Operasi Dasar 8255 .....	29
2.5. Pemilihan saluran input .....	46
3.1. Tabel kebenaran address decoding PPI satu..	
3.2. Tabel kebenaran address decoding PPI dua ..	
4.1. Tabel Output tegangan.....	
4.2. Tabel Output frekwensi.....	



# BAB I

## PENDAHULUAN

### 1.1 LATAR BELAKANG

Dunia Industri sebagai suatu kelompok besar pengguna energi listrik, banyak menggunakan energi listrik sebagai energi utama dalam menggerakkan peralatan - peralatannya (misalkan motor - motor ). Suplai energi listrik dalam industri biasanya didapatkan dari PLN (Perusahaan Listrik Negara) sebagai sumber utama dan generator sebagai sumber pembantu dan kadang - kadang dioperasikan sebagai cadangan. Pada beban maksimum kadang - kadang diperlukan suplai pembantu untuk mendukung fungsinya agar tidak mengalami kelebihan beban (overloaded). Penambahan suplai dari generator untuk bekerja bersama diperlukan untuk menanggung beban sehingga tidak mengganggu jalannya proses produksi.

Perkembangan mikroelektronika dan komputer yang pesat dapat diaplikasikan dalam industri dalam hal ini digunakan untuk membantu pengaturan sistem kerja paralel generator, agar didapatkan suatu peralatan yang handal dan efisien.

Pemakaian teknologi mikroelektronika dan komputer didasarkan pada pertimbangan keandalan, efisiensi kerja dan kemudahan pengoperasian dibandingkan bila menggunakan peralatan mekanis dan manual yang selama ini dipakai.

Setelah mengetahui keandalan dengan menggunakan teknologi mikroelektronik dan komputer maka diharapkan didapatkan hasil yang optimum dari peralatan industri yang



MILIK PERUSAHAAN  
INSTITUT TEKNOLOGI  
SEPULUH - NOPEMBER

dipakai.

## 1.2 PERMASALAHAN

Judul tugas akhir yang diambil adalah : Pengaturan sinkronisasi Generator tiga phasa yang diinterfacekan ke IBM PC XT. Permasalahan sinkronisasi generator tiga phasa adalah pemenuhan persyaratan sinkronisasi sebelum dilakukan kerja paralel generator.

Syarat - syarat kerja paralel generator harus dipenuhi lebih dahulu untuk melakukan kerja paralel karena bila tidak dipenuhi akan terjadi kerusakan pada generator yang akan dioperasikan paralel.

Tujuan dari tugas akhir ini adalah merencanakan dan membuat suatu peralatan yang dapat mengoperasikan sinkronisasi generator pada kerja paralel yang selama ini dilakukan dengan operasi manual dan mekanis. Adapun permasalahan yang dibahas dan dipelajari adalah :

1. Pemanfaatan Unit Input dan Output (I/O Unit ) sebagai jalur keluar dan masuknya data dari dan ke IBM PC atau disebut Interfacing ke IBM PC.
2. Proses pengambilan dan konversi data dari rangkaian analog ke digital dan sebaliknya dari peralatan kontrol yang mendeteksi besaran - besaran syarat kerja paralel dari generator.
3. Rangkaian Analog peralatan kontrol yang akan mendeteksi data - data syarat - syarat generator paralel.

## 1.3 PEMBATASAN MASALAH

Dalam tugas akhir ini dilakukan perencanaan dan pembuatan peralatan kontrol sinkronisasi generator tiga

phasa yang diinterfacekan dengan IBM PC XT. Digunakannya generator tiga phasa adalah merupakan peralatan yang mendekati keadaan dalam industri. Pada pengontrolan menggunakan IBM PC XT digunakan software/atau programming Turbo Pascal untuk mengatur peralatan sinkronisasi.

Dari masing - masing Unit akan dihubungkan secara hardware ke IBM PC XT dan nantinya IBM PC XT akan memberikan keputusan saat yang tepat untuk melakukan kerja paralel.

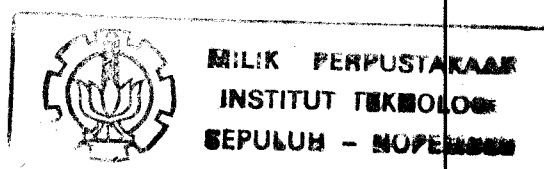
#### 1.4 SISTEMATIKA PEMBAHASAN

Pada tugas akhir ini dilakukan pembahasan dengan sistematika berikut ini,

- Bab I membahas tujuan pemilihan judul tugas akhir
- Bab II membahas teori penunjang berupa teori dasar mikroprosesor dan peralatan penunjangnya, yaitu I/O unit dan peralatan analognya. Juga dibahas mengenai peralatan utama yaitu generator dan motor yang digunakan untuk pengoperasian kerja paralel.
- Bab III membahas dasar perencanaan rangkaian dan komponen yang digunakan serta pembuatan software sistem untuk mengontrol sistem secara keseluruhan.
- Bab IV Pengukuran dan pegujian dari alat guna melengkapi bukti kebenaran dari perencanaan alat tersebut.
- Bab V merupakan kesimpulan dari seluruh pembahasan Tugas Akhir.

#### 1.5. METODOLOGI

Pembahasan yang dilakukan bersifat praktis dan teoritis. Secara teoritis akan dibahas peralatan pengontrol dan Central Prosesing Unit yang digunakan beserta komponen -



komponen yang menunjangnya. Secara praktis akan diuraikan mengenai pembuatan alat secara riil sesuai dengan tugas yang diberikan untuk Tugas Akhir ini.

## BAB II

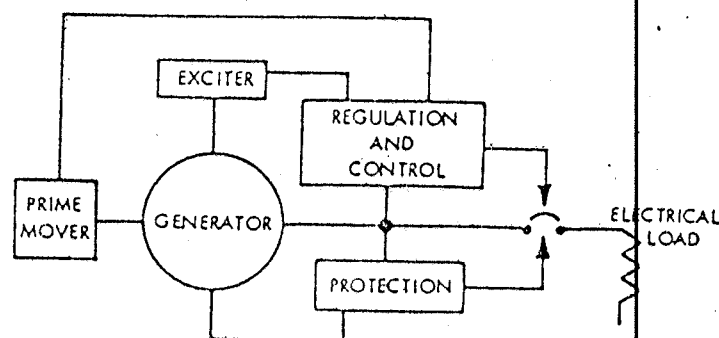
# TEORI PENUNJANG

### II.1. SISTEM PEMBANGKIT TENAGA LISTRIK

Di dalam sistem pembangkit Tenaga listrik terdapat bagian - bagian atau komponen sebagai berikut :

1. Generator
2. Penggerak Utama
3. Pembangkit Medan
4. Perlengkapan Pengatur
5. Perlengkapan Pelindung

Kemudian sistem Pembangkit listrik dapat digambarkan dalam bagan dalam gambar 2.1 di bawah ini.



Gambar 2.1 1)

Bagan Sistem Pembangkitan Listrik

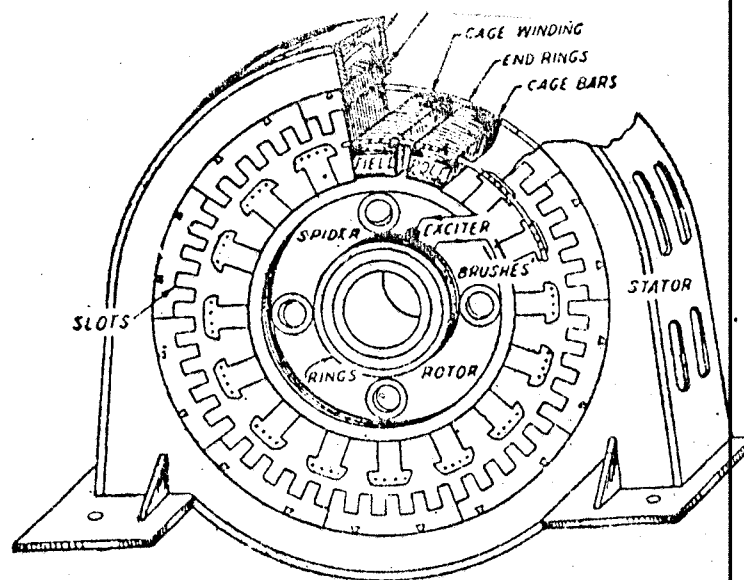
#### II.1.1 GENERATOR

Generator adalah peralatan yang dapat mengubah suatu energi mekanik menjadi energi listrik dengan memakai rotor sumbu penggerak utama (prime mover), dimana perubahan tersebut didasarkan pada prinsip induksi EMF.

1) Basler Electric, 1989 Control of Electric generating System  
Hygland - Illinois, USA hal 1



Generator terdiri dari dua bagian, yaitu Rotor dan Stator. Stator atau bagian yang tidak bergerak terdiri dari lilitan daya tiga fasa atau lilitan jangkar. Rotor atau bagian yang bergerak mempunyai kutub medan magnetik yang dibangkitkan oleh arus searah. Pada saat konduktor memotong flux magnetik maka akan menghasilkan induksi dinamis EMF yang sesuai dengan Hukum Faraday tentang Induksi Electromagnetic. Ada dua macam jenis generator yaitu generator serempak (synchronous generator) dan generator tak serempak (asynchronous generator).



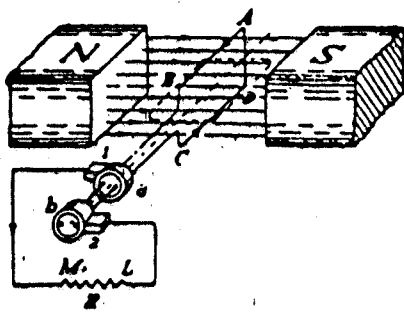
Gb. 2.2 2)

#### Gambar Konstruksi Generator

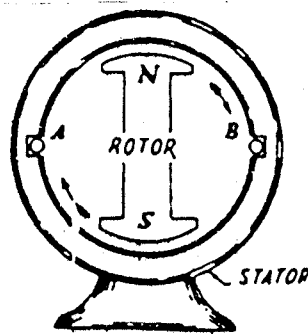
Pada pemakaian praktek sehari - hari generator serempak lebih banyak digunakan karena kemudahan dalam pengaturan tegangan, frekwensi, daya dan daya reaktifnya.

Stator merupakan bagian yang tidak berputar terdiri dari besi bingkai cor yang menyangga besi jangkar, memiliki slots - slots pada bagian dalamnya untuk menempatkan tahanan jangkar. Sedangkan rotor seperti roda gerigi yang mempunyai kutub Utara (N) dan Selatan (S) dibagian luarnya. Kutub Magnetis di exitasi (dimagnetisasi) dengan arus DC yang disuplai oleh sumber DC pada 2 - 4 Ampere.

Karena medan magnet itu berputar, arus disuplai melalui dua cincin belah. Sebagai tegangan existing yang relatif kecil maka ( cincin belah ) dan sikat dibuat dengan konstruksi yang sederhana. Jika rotor berputar, tahanan stator akan dipotong oleh fluks magnetik dan akan menginduksi Emf yang dihasilkannya. Karena kutub magnetik terdiri dari kutub U dan kutub S, akan menginduksi emf dan arus di tahanan jangkar. Dimana aliran pertamanya akan searah. Dalam hal ini emf bolak balik dihasilkan dalam penghantar stator yang



b. 2.3.3.a)



Gb. 2.3.3.b)

#### Proses Pembangkitan Listrik

frekwensinya tergantung oleh jumlah kutub U dan S nya yang melewati penghantar dalam tiap detiknya dan arahnya sesuai dengan hukum tangan kanan Fleming.

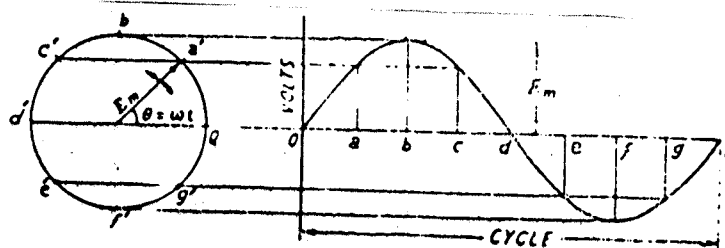
### II.1.2 PEMBANGKITAN ARUS DAN TEGANGAN.

Jika dimisalkan suatu lilitan mempunyai  $N$  lilitan dan berputar pada suatu bidang magnet yang sama dan memiliki kecepatan sudut angular  $\omega$  rad/det dalam waktu  $t$  detik maka : menurut hukum Faraday bahwa EMF terinduksi adalah diberikan dengan persamaan sebagai berikut :

$$\begin{aligned}
 e &= -\frac{d}{dt} (N \Phi) \text{ volt} \\
 &= -N \frac{d}{dt} (\Phi_m \cos \omega t) \text{ volt} \\
 &= -N \Phi_m \omega (-\sin \omega t) \text{ volt} \\
 &= N \Phi_m \omega \sin \omega t \text{ Volt} \\
 &= \omega N \Phi_m \sin \omega t \text{ Volt} \\
 &= \omega N \Phi_m \sin \theta \text{ Volt} \dots \dots (1)
 \end{aligned}$$

Dimisalkan  $\theta = 90^\circ$  atau  $e$  berharga maksimum maka

$$\begin{aligned}
 E_m &= \omega N \Phi_m = \omega N B_m A \\
 &= 2\pi f N B_m A \text{ Volt} \dots \dots (11)
 \end{aligned}$$



Gambar 2.4 <sup>4)</sup>

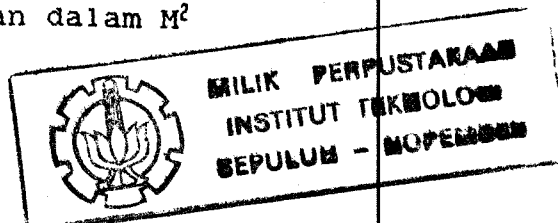
Bentuk gelombang yang dibangkitkan

dimana

$B_m$  = Kerapatan Fluks dalam  $\text{Wb/m}^2$

$A$  = luas lilitan dalam  $\text{M}^2$

<sup>4)</sup> Ibid, hal 322



maka jika persamaan tersebut digabungkan,

$$\begin{aligned} e &= E_n \sin \theta \\ &= E_n \sin \omega t \end{aligned} \quad \dots\dots (3)$$

Persamaan tersebut identik pula untuk arus

$$i = I_n \sin \omega t$$

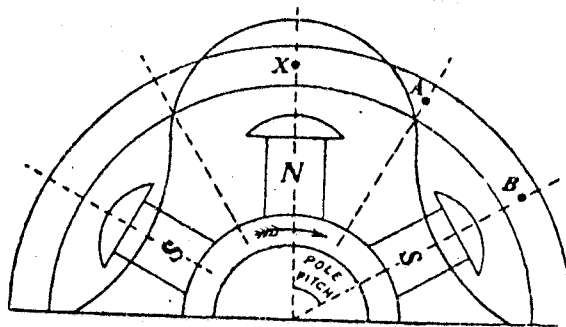
Jika  $\omega = 2\pi f t$ , maka

$$\begin{aligned} e &= E_n \sin 2\pi f t \\ &= E_n \sin \frac{2\pi}{T} t \\ i &= I_n \sin \frac{2\pi}{T} t \end{aligned}$$

Dapat disimpulkan dari persamaan diatas bahwa emf terinduksi merupakan suatu gelombang sinus.

### II.1.3 KECEPATAN DAN FREKWENSI

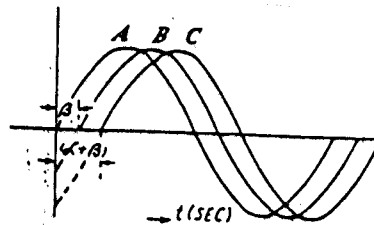
Dalam generator AC terdapat hubungan antara kecepatan putar (N) rotor dengan frekwensi emf yang dihasilkan dan jumlah kutub.



Gambar 2.5 5)

Medan magnetik rotor yang berputar

Dimisalkan penghantar jangkar yang diberi tanda x terletak di tengah - tengah dari kutub N yang berputar searah jarum jam, maka penghantar terletak pada tempat yang mempunyai

Gambar 2.6<sup>6</sup>

Kurva tegangan dan waktu

Kurva A mempunyai persamaan

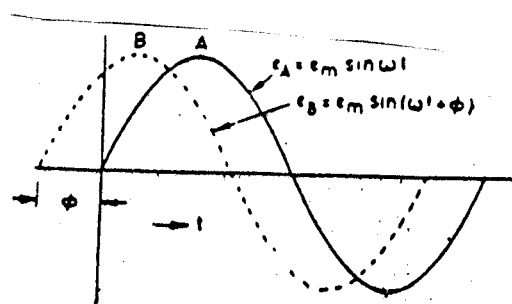
$$e_A = E_m \sin Wt$$

Kurva B mempunyai persamaan

$$e_B = E_m \sin ( Wt - \beta )$$

Kurva C mempunyai persamaan

$$e_C = E_m \sin ( Wt - ( \alpha + \beta ) )$$

Gambar 2.7<sup>7</sup>

Kurva fasa "lead" atau "lag"

6) Ibid, hal 326

7) Ibid, hal 327

kerapatan flux yang maximum akan mempunyai emf yang maximum yang diinduksikan pada penghantar. Jika penghantar terletak di titik A maka mempunyai emf terinduksi yang minimum dan jika terletak di tengah kutub S maka mempunyai emf terinduksi yang maksimum karena kerapatan fluks di titik B adalah maksimum. Misalkan

$P$  = Jumlah kutub magnetik

$N$  = Kecepatan rotasi

$f$  = Frekuensi GGL induksi dalam Hertz

maka cycle/putaran penuh =  $P/2$

Jumlah putaran penuh/detik =  $n/60$

Frekwensi yang dihasilkan =  $P/2 \times n/60$

Jadi didapatkan rumus  $f = \frac{PN}{120}$  Hz

Jadi untuk tegangan bolak - balik dengan frekwensi 50 hz, dibangkitkan dengan generator yang mempunyai 2 pasang kutub dibutuhkan kecepatan putar dari penggerak utamanya 1500 rpm.

#### II.1.4 PERBEDAAN PHASA

Jika dimisalkan terdapat tiga lilitan tunggal yang hampir sama satu sama lain dan dipisahkan dengan sudut dan diputar dalam medan magnet yang sama dan diputar dengan kecepatan sudut yang sama, maka nilai emf terinduksi ketiga lilitan tersebut adalah sama tetapi emf dari ketiga lilitan tersebut tidak mencapai maksimum atau minimum secara bersamaan, maka dapat dilihat pada gambar 2.6

Jika B mendahului A dengan sudut  $\phi$  seperti dalam gambar diatas maka persamaannya adalah :

$$e_A = E_m \sin \omega t$$

$$e_B = E_m \sin (\omega t + \phi)$$

Tanda + adalah mendahului dan - adalah tertinggal

## II.2 PENGGERAK UTAMA

Penggerak utama generator dapat berupa mesin diesel, turbin uap atau gas, turbin air ataupun turbin angin. Dalam tugas akhir ini digunakan motor sebagai penggerak utama generator.

### II.2.1 MOTOR SEBAGAI UNIT PENGGERAK

Pada prinsipnya untuk menggerakkan suatu peralatan dengan arah rotasi maupun translasi dibutuhkan sumber penggerak dalam hal ini digunakan : motor listrik, yang kerjanya sebagai pengubah energi listrik menjadi energi gerak (energi mekanis). Adapun pemilihan motor sebagai penggerak harus memperhatikan : kecepatan, pengaturan Kecepatan, torsi motor, daya motor dan sebagainya. Hal tersebut untuk mendapatkan efisiensi kerja yang tinggi.

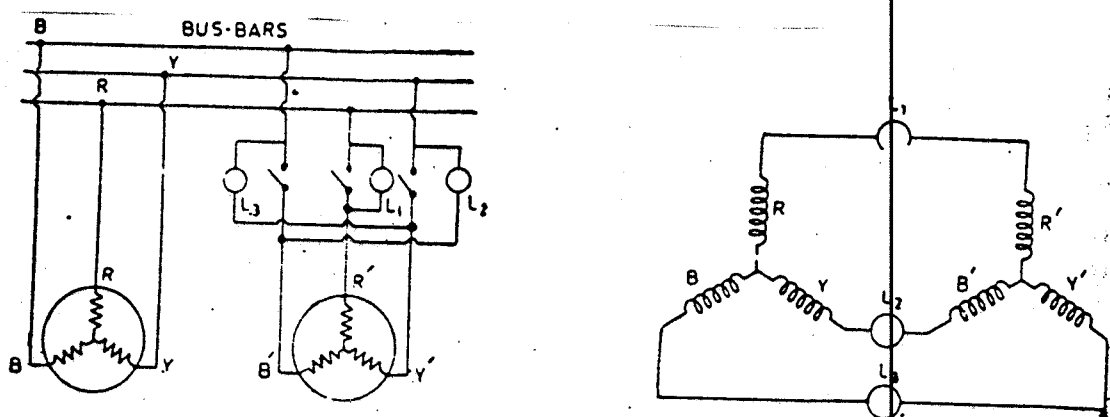
## II. 3. KERJA PARALEL GENERATOR.

Untuk melakukan kerja paralel maka perlu dipenuhi kondisi - kondisi berikut :

- (1). Tegangan efektif terminal harus sama besar dengan dengan tegangan genarator lain
- (11). Kecepatan dari generator yang akan diparalelkan dalam hal ini adalah frekwensinya harus sama dengan frekwensi

generator lain.

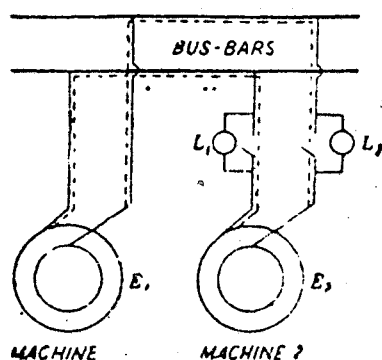
- (iii). Fasa dan urutan fasa tegangan generator harus identik dengan fasa tegangan generator lain.



Gambar 2.8 <sup>8)</sup>

#### Hubungan kerja paralel antar Generator

Untuk mendeteksi kerja paralel tersebut kita menggunakan syncronscop lampu yang menggunakan prinsip hubungan terang atau gelap untuk mendeteksi frekwensi dan urutan fasa dari generator yang akan dihubungkan paralel



Gambar 2.9 <sup>9)</sup>

#### Hubungan antar Busbar

<sup>8)</sup> Ibid, hal 1015

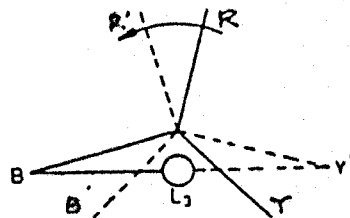
<sup>9)</sup> Ibid, hal 1016



Sedangkan voltmeter digunakan untuk mendeteksi tegangan.

Jika sebuah generator akan bekerja paralel dengan generator lainnya maka Generator tersebut diputar terlebih dulu mendekati kecepatan sinkronnya dengan sebuah motor penggerak mula dan diatur  $I_f$  ( arus medannya ) sehingga tegangan terminal generator satu sama dengan generator lainnya. Sedangkan untuk mendeteksi frekwensi dan beda fasa antar generator dan digunakan hubungan lampu synchroscop hubungan gelap dan terang. Jika rangkaian tersebut benar artinya urutan fasanya benar maka lampu - lampu tersebut akan berkedip sesuai dengan selisih frekwensi listrik antar generator. tetapi jika lampu - lampu tersebut tidak berkedip maka frekwensi listrik sudah sama antar generator. Lampu - lampu tersebut diletakkan pada tiap sudut suatu bentuk segitiga dimana kecepatan berkedipnya lampu tersebut akan menunjukkan kecepatan berputarnya generator. Saat memparalelkan adalah pada saat L1 mati, L2 dan L3 menyala terang. Dalam keadaan ini posisi semua fasa sistem antar generator saling berimpit.

Untuk mensinkronisasi generator tiga fasa maka hanya diperlukan mensinkronisasi satu fasa saja, karena fasa yang lainnya akan mengikuti secara otomatis



Gambar 2. 10<sup>10</sup>)  
Diagram Phasor antar fasa

## II. 4. ARSITEKTUR MIKROKOMPUTER IBM PC

IBM PC atau International Bussiness Machine Personal Computer adalah salah satu produk komputer yang tergolong sebagai mikrokomputer yang sejak satu dekade lebih mengalami perkembangan teknis dan pemasaran yang sangat pesat. Hal ini memang karena IBM PC sangat luwes dalam aplikasinya dan tidak memonopoli dalam menciptakan suatu unit sistem serta pengembangan sistem operasinya, melainkan bekerjasama dengan perusahaan lainnya seperti Microsoft, Seattle Computer Company, Sistem Operasi Digital Research, Tandon Magnetig, Epson serta yang lainnya. <sup>11)</sup>

Konfigurasi sistem dasar IBM PC meliputi unit sistem perangkat elektronik utama komputer, dua diskdrive, keyboard, sebuah tampilan ( video display ) dan printer, yang tergambar pada gambar 2.7. Di dalam unit sistem IBM PC yang dipakai terdapat microprosesor 16 bit 8088 yang bekerja sebagai unit pemroses utama.

Di dalam Unit Sistem IBM PC yang dipakai terdapat microprosesor 16 bit 8088 yang bekerja sebagai unit pemroses utama, read only memory, random access memory, Interface untuk diskdrive dan peralatan monitor, catu daya, speaker untuk aplikasi audio, dan delapan ekspansi slot untuk pengembangan lebih lanjut.

---

11) Breiner C. Robert, IBM PC Trouble shooting & Repair Guide  
( Slawson Communication, Inc )

Gd. 2.11<sup>12</sup>

Diagram IBM - PC XT

## II. 5. MIKROPROSESOR 8088

Mikroprosesor 8088 adalah merupakan pengembangan mikroprosesor 8086 yang diproduksi oleh INTEL. terdapat perbedaan pokok diantara mikroprosesor tersebut, adalah terletak pada lebar data busnya.

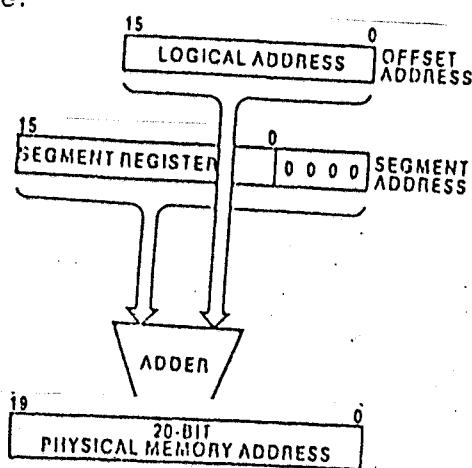
---

<sup>12)</sup> Lewis. C. Eggebrecht, Interfacing to IBM Personal Computer

Indianapolis, Indiana Howard M. Sams & Co, Inc, 1983), Hal 21

### II.5.2. PENGADDRESSAN PADA MEMORY

Didalam 8088 mikroprosessor mempunyai karakteristik khusus, yaitu mampu memberikan address lebih dari 65.536 byte data yang ditentukan oleh 16 bit address field. Mikroprosessor 8088 mempunyai 20 bit address yang secara fisik mampu memberikan address pada memory sebesar 1.048.576 byte. Akan tetapi instruksi - instruksi pada 8088 harus disesuaikan dan dimanipulasi pada address 16 bit. Sehingga hanya memory 65.536 byte saja yang digunakan oleh pemrogram. Untuk mengatasi hal ini lokasi memory sebesar 1 megabyte tersebut dibagi menjadi segment - segment yang berukuran 64 Kbyte saja yang lokasi awalnya ditunjuk oleh register 16 bit khusus yang disebut segment register. Address pada segment register digunakan pada lokasi dalam daerah 1 megabyte dengan menggunakan instruksi pada 8088 yang beroperasi pada daerah Untuk mencegah masalah ini, segment register 16 bit ditentukan pada batasan 16 byte.



GAMBAR 2. 12<sup>13</sup>

PEMBANGKITAN ADDRESS MEMORY PADA 8088

Pada 8086 pin datanya sebanyak 16 pin yang sebagian pin tersebut terhubung dengan pin address. 8086 dikelompokkan dalam mikroprosesor 16 bit karena memiliki jalur data internal dan eksternal masing - masing 16 bit. Lain halnya dengan 8088 yang memiliki jalur data internal 16 bit dengan jalur data eksternal yang hanya 8 bit. Akan tetapi 8088 dikelompokkan dalam mikroprosesor 16 bit karena kemampuannya untuk melaksanakan operasi yang menggunakan operan 16 bit.

Intel 8088 beroperasi pada frekwensi 4,77 Mhz dan bisa diperbesar sampai 10 Mhz.

Untuk lebih jelasnya bisa dilihat pada jalur - jalur yang ada pada mikroprosesor 8088 CPU.

#### II.5.1. FUNGSI PADA KAKI - KAKI IC 8088 CPU

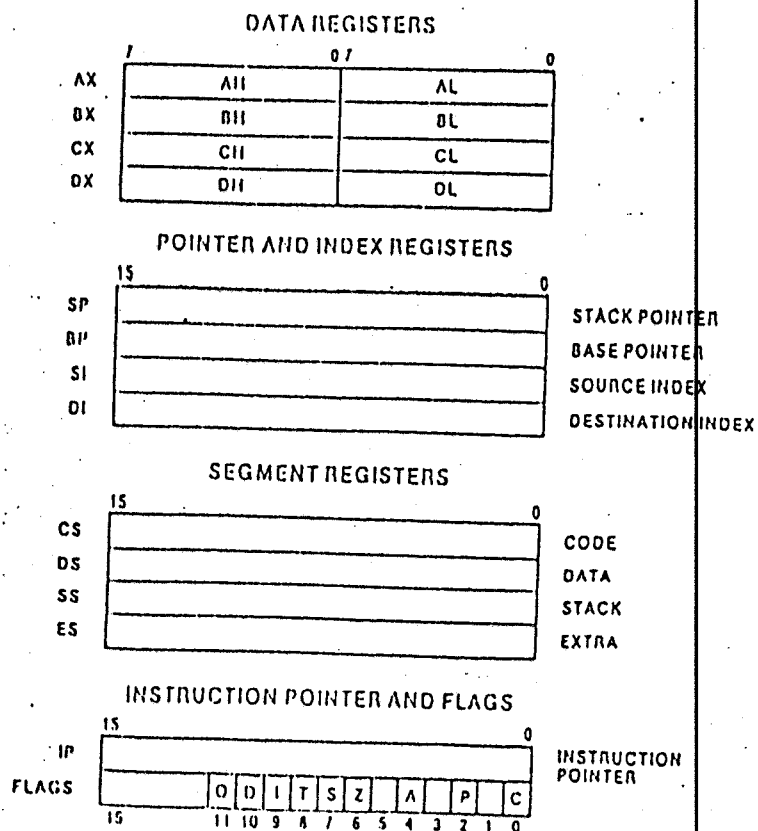
Pada IC 8088 terdapat 40 kaki yang bisa terbagi menjadi delapan kelompok dari kaki - kaki yang ada pada 8088 ialah :

1. 8 bit untuk sistem data
2. 20 jalur untuk address bus
3. Jalur untuk sumber tegangan dan ground untuk interface board.
4. Jalur clock dan sinyal - sinyal timer.
5. Jalur Control untuk memory atau I/O read atau write
6. Jalur maskable dan non maskable interupt request.
7. Jalur untuk status flag.
8. Jalur bus cycle status

Dalam 1 megabyte, pada beberapa batasan 16 byte, physical memory address dibentuk oleh penambahan isi segment register 4 bit ke kiri dan penambahan itu dibentuk dari instruksi address 16 bit. Seperti pada gambar 2 - 1 diatas.

Mikroprosesor 8088 mempunyai empat belas register 16 bit yang terbagi dalam beberapa register antara lain :

- Empat data register
- Empat pointer dan indeks register
- Empat segment register
- Satu instruction pointer register
- Satu flag register



Gambar 2.13

Register 8088

### II.5.3 DATA REGISTER

Kelompok data register terdiri dari empat register enam belas bit yang biasa digunakan oleh instruction set untuk melaksanakan operasi - operasi logika dan aritmatika. Keempatnya adalah sebagai berikut :

- AX (accumulator)
- BX (base register)
- CX (count register)
- DX (data register)

Register - register ini juga dapat di address sebagai register - register 8 bit ( AH, AL, BH, HL, CH, CL, DH dan DL) untuk operasi - operasi byte dari instruction set.

### II.5.4 POINTER DAN INDEKS REGISTERS

Kelompok pointer dan indeks register terdiri atas empat register 16 bit yang digunakan untuk membentuk efektif memory address. Untuk ke empatnya adalah sebagai berikut :

- SP ( Stack Pointer )
- BP ( Base Pointer )
- SI ( Source Indeks )
- DI ( Destination Indeks )

Register - register digunakan untuk operasi logika dan aritmatika.

### II.5.5 SEGMENT REGISTER

Kelompok register segment terdiri atas empat register 16 bit ( CS, DS, SS, ES ) yang digunakan untuk menempatkan segment - segment 64 Kbyte di dalam lokasi address 1 megabyte. Bagian dari address awal dari segment

register yang disimpan dalam segment register disebut segment base. Keempat register tersebut adalah :

- CS ( code segment )
- DS ( Data segment )
- SS ( stack segment )
- ES ( Extra segment )

#### II.5.6 INSTRUCTION POINTER REGISTER

Instruction register merupakan (IP) register merupakan sebuah register 16 bit yang memuat offset address dari instruksi berikutnya, ditinjau terhadap address base dari code segment yang sekarang.

#### II.5.7 FLAG REGISTER

Flag register merupakan sebuah register 16 bit tetapi hanya sembilan bit - nya saja yang digunakan seperti yang tampak pada gambar 2 - . Enam dari bit - bit ini merupakan bit - bit status yang mencerminkan hasil operasi - operasi logika dan aritmatika sebagai berikut :

- Bit 0, Carry flag (CF)
- Bit 2, Parity flag (PF)
- Bit 4, Auxilliary Carry Flag (AF)
- Bit 6, Zero flag (ZF)
- Bit 7, Sign flag (SF)
- Bit 11, Over flow flag (OF). Ketiga bit yang lain merupakan bit-bit kontrol, sebagai berikut :
- Bit 8, trap flag (TF)
- Bit 9, Interrupt Flag (IF)
- Bit 10, Direction Flag (DF)



## II. 6. RANGKAIAN CLOCK

Pada mikrokomputer IBM PC - XT, mikroprosesor 8088 bekerja dengan clock 4,77 Mhz. Rangkaian oscillator menggunakan crystal 14,31818 Mhz yang dibagi tiga oleh IC 8284 selanjutnya sinyal tersebut dibagi empat menghasilkan sinyal 1,19 Mhz untuk memberi sinyal clock pada timer/conter

## II.7 SISTEM BOARD BUS

Bagian - bagian yang fungsional sistem board IBM PC dihubungkan ke CPU 8088 melalui sistem bus. Bus ini terdiri bermacam - macam antara lain data, adres, control, timing, interrupt request dan DMA control. Sstem board juga dilengkapi dengan 8288 bus controller chip yang menghasilkan sinyal - sinyal control.

## II.8. SISTEM MEMORY

Pada sistem board IBM PC digunakan 2 macam memory RAM dan ROM. RAM adalah Random Access Memory dimana memory ini akan hilang bila listrik dimatikan sedangkan ROM adalah Read Only Memory atau memori yang bersifat resident, untuk pembagian lokasi memory serta fungsinya dalam IBM PC dapat dilihat pada tabel 2.1

RAM pada lokasi address 0000H sampai 0BFFFFH dapat dilihat pembagiannya lebih terinci dalam tabel 2 - 6. Disini bisa dilihat bahwa address 0000H sampai address 03FFFH digunakan ROM berada menurut fungsinya dibagi seperti dalam tabel 2 - 1 Dalam tabel 2 - 2. Dibawah ini tampak bahwa lokasi RAM address 00000H sampai 003FFFH dipakai untuk penyimpanan vektor interrupt dari program yang terdapat pada BIOS, program DOS, program BASIC atau program lainnya.

00400H 004FFH	Daerah data BIOS
00500H 005FFH	Daerah data BASIC dan DOS
00600H 006FFH	62,5 K RAM untuk pemakai

Pemakaian lokasi address oleh ROM terletak pada address C0000H sampai FFFFFH yang dialokasikan pada 262.144 atau 256K lokasi memory yang dapat mengatur unit input dan output yang digambarkan pada tabel dibawah.

TABEL 2.3 PEMBAGIAN LOKASI MEMORY ROM

Address ( Hex )	system 16K/64K	system 64K/256K
C0000H C7FFFH	32K	192K ROM untuk ekspansi dan kontrol
C8000H CBFFFH	16K Hard disk control	
CC000H F3FFFH	tidak dipakai	
F4000H F5FFFH	Soket untuk ROM	
F6000H FDFFFH	ROM BASIC (32K)	
FE000H FFFFFH	ROM BIOS (8K)	

## II.9 SYSTEM BOARD TIMER /COUNTER

Untuk mendukung fungsi timing/counting, IBM PC - XT dilengkapi dengan sebuah IC 8253 Timer/counter dengan tiga timer/counter 16 bit. Untuk input clock dari ketiga channel timer/counter ini di drive oleh sinyal clock 1,19 Mhz. Output timer channel 0 dihubungkan dengan sistem interrupt level 0, dan diprogram untuk membangkitkan interrupt setiap

54,925 millidetik ( sekitar 18,2 kali per detik ). Timer ini di gunakan oleh routine - routine I/O dan penunjukkan waktu clock.

Output timer Channel 1 digunakan untuk membangkitkan sinyal DMA request pada DMA channel 0, yang digunakan untuk menyegarkan memory dinamik pada sistem. Hal ini dilakukan dengan menghasilkan suatu dummy memory read cycle setiap 72 clock prosessor ( atau setiap 15,12 microdetik ).

Output timer channel 2 digunakan untuk mendrive audio speaker dan mentransmisi data ke audio cassette port pada sistem board.

## II.10 SISTEM BOARD DMA

Beberapa peralatan I/O seperti diskette drive, mentransmisi data lebih cepat daripada yang dapat di tangani oleh CPU dibawah control suatu program. Untuk mengatasi hal ini, sistem board IBM PC XT dilengkapi dengan 8237 DMA controller. IC ini mengirimkan data antara memory dan peralatan adapter tanpa menyertakan CPU.

IC controller ini memiliki empat DMA channel yang memiliki fungsi sebagai berikut. ( berurutan dari prioritas tertinggi sampai yang terendah ).

- Channel 0 : digunakan untuk menyegarkan RAM
- Channel 1 : belum digunakan
- Channel 2 : digunakan oleh diskette drive adapter
- Channel 3 : belum digunakan

## II.11 SISTEM BOARD INTERRUPT

Mikroprosessor 8088 mempunyai dua fasilitas interrupt hardware, yaitu maskable interrupt dan non maskable

interrupt. Karena level interrupt yang diperlukan lebih banyak daripada yang tersedia, maka sistem board IBM PC XT dilengkapi dengan 8259 Interrupt Priority Controller. Dengan peralatan ini kemampuan masakable interrupt diperluas sampai delapan level interrupt (IRQ1 - IRQ8).

Penggunaan non maskable dan masakable interrupt pada IBM PC - XT adalah sebagai berikut ( berurutan dari prioritas tertinggi sampai terendah ) : FF)

- NMI : Base board RAM parity, I/O channel check, untuk numerik prosessor 8087
- IRQ0 : System timer output 8253 channel 0
- IRQ1 : Keyboard scan code interrupt
- IRQ2 : belum digunakan
- IRQ3 : asynchronous & SDLC adapter (sekunder)
- IRQ4 : asynchronous & SDLC adapter ( primer )
- IRQ5 : fixed - disk adapter
- IRQ6 : disket drive adapter
- IRQ7 : paralel printer adapter

Pada slot tambahan di IBM PC - XT dapat di jumpai input - input IRQ2 sampai IRQ7 yang dapat digunakan oleh user, asalkan belum digunakan oleh adapter - adapter di atas atau keperluan yang lain

## II.12 SYSTEM BOARD INTEGRATED I/O ADAPTER

Sistem board IBM PC XT di lengkapi dengan adapter - adapter untuk keyboard, audio speaker, dan audio cassette port. Peralatan - peralatan dipadukan pada sistem bus dengan menggunakan 8255. Chip ini memiliki tiga port I/O 8 bit.

Port - port ini digunakan dalam hal - hal sebagai berikut :

- Membaca data dari dip switch konfigurasi system
- Membaca scan code dari keyboard
- mendrive audio speaker
- Membaca serial data
- mengontrol motor pada disk drive
- dll

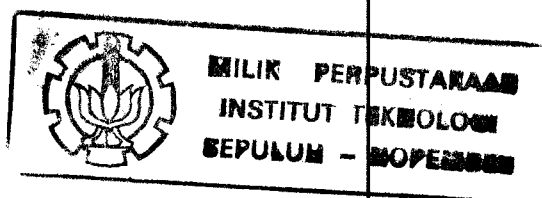
## II.13      PROGRAMMABLE      PERIPHERAL      INTERFACE      8255

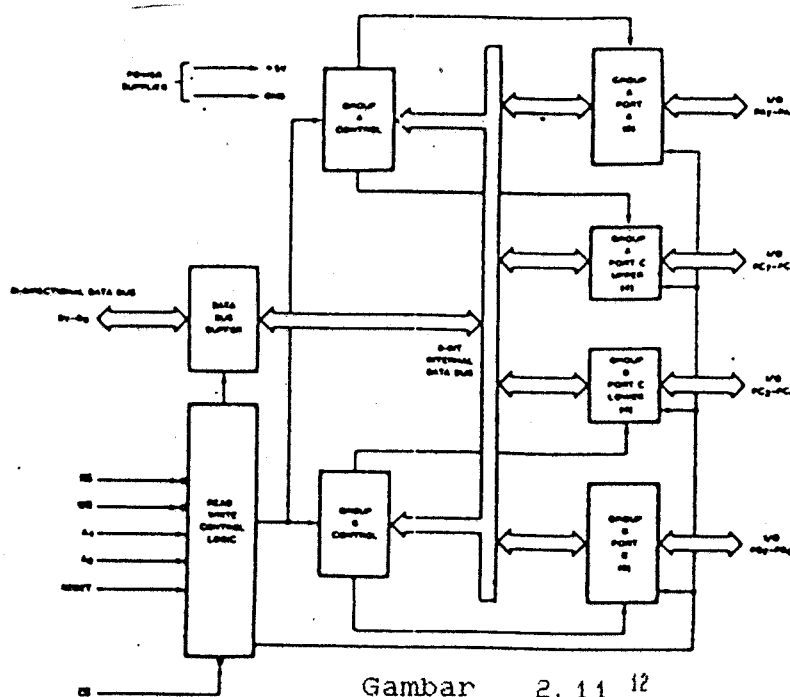
Programable Peripheral Interface 8255 adalah rangkaian terpadu ukuran besar yang dikemas dalam dual in line yang dirancang untuk digunakan menginterface bermacam - macam fungsi I/O pada sistem Mikroprosesor (CPU).

Didalam diagram 8255 terdapat empat block yang secara fisik dapat dihubungkan dengan peralatan di luar. Empat blok tersebut adalah :

- Port A (PA0 - PA7)
- Port B (PB0 - PB7)
- Port C lower (PC0 - PC3)
- Port C upper (PC4 - PC7)

Kelompok sinyal tersebut terbagi menjadi tiga I/O port yang berbeda yaitu port A, port B dan port C. Semua bagian di dalam 8255 tersebut dihubungkan dengan internal bus data, dan melalui internal bus data inilah data dikirim dan diterima oleh setiap port.





Gambar 2.11 12  
Blok diagram PPI 8255.

#### II.14. FUNGSI PIN - PIN PADA 8255

Pada bagian ini kita menerangkan pin - pin yang terdapat pada 8255, dan fungsi - fungsinya.

##### 1) $D0 - D7$

Merupakan input dan output dari PPI 8255. Semua data akan dibaca dan ditulis dari dan ke PPI 8255 melalui delapan jalur ini.

##### 2) $\overline{CS}$

$\overline{CS}$  aktif rendah maka PPI 8255 dapat berhubungan dengan CPU.

##### 3) $\overline{RD}$

$\overline{RD}$  aktif rendah, dengan  $\overline{CS}$  berlogika rendah maka 8255

memungkinkan untuk mengirim data ke CPU, sehingga dapat dikatakan CPU melakukan pembacaan data dari 8255.

- 4)  $\overline{WR}$   
Input  $\overline{WR}$  aktif rendah, dengan CS berlogika rendah juga maka 8255 memungkinkan menerima data dari CPU atau menerima control word dari CPU.

- 5) A0 - A1

Kombinasi dari kedua jalur alamat ini dapat menentukan pemilihan salah satu dari 3 port dan 1 control word register.

- 6) RESET

Fungsi dari pin ini adalah untuk mereset PPI dengan sinyal tinggi. Pada saat reset ini mengakibatkan control word register terhapus dan ketiga port diset sebagai mode input.

- 7) PA0 - PA7 (Port A)

Digunakan sebagai I/O yang dapat dihubungkan dengan peralatan luar.

- 8) PB0 - PB7 (Port B)

Digunakan sebagai I/O tepat sama dengan port A, akan tetapi kedua port ini saling bebas satu dengan lainnya.

- 9) PC0 - PC7 (Port C)

Digunakan sebagai I/O sama halnya dengan port A atau port B akan tetapi hanya ada perbedaan. Perbedaan port C ini dengan port yang lainnya adalah pembagian port C upper (PC4 - PC7) dan port C lower (PC0 - PC3). Port C upper bisa digunakan untuk mengendalikan port A sedangkan port C lower dapat digunakan untuk mengendalikan port B.

## II.14.1 CARA MENGHUBUNGKAN 8255 DENGAN MIKROPROSESSOR

Untuk menghubungkan 8255 dengan mikroprosesor maka hubungan dilakukan dengan menghubungkan pin Address mikroprosesor dengan pin address 8255, pin data dengan pin data, IOW dan IOR CPU dengan WR dan RD PPI, karena pin RD dan WR dari CPU tidak dapat langsung dihubungkan dengan RD dan WR dari PPI karena jalur ini akan aktif pada waktu pengoperasian memory dan untuk itu diperlukan pengalamatan dari address addressnya, sedangkan pengalamatan ditentukan dengan kombinasi antara A0 dan A1 sedangkan port address khusus dikontrol dengan pin CS ( chip select ) dimana bila aktif harus sama dengan nol (low) dan harus diperhatikan pula I/O mapping pada mikroprosesor.

Pada PPI 8255 ini mempunyai empat internal register, yaitu register - register PA, PB, PC dan Control word. Keempat internal register tersebut dapat

Tabel 2. 6 : Operasi Dasar 8255

RD	WR	A1	A0	operasi
1	0	0	0	Write port A
0	1	0	0	Read port A
1	0	0	1	Write port B
0	1	0	1	Read port B
1	0	1	0	Write port C
0	1	1	0	Read port C
1	0	1	1	control word
0	1	1	1	kondisi terlarang



diakses sehingga dapat dioperasikan sebagai unit Input dan output dengan mengatur hubungan antara pin - pin A0, A1, WR dan RD seperti yang ditunjukkan dalam tabel diatas.

8255 mempunyai tiga mode operasi yang dapat diatur dengan mengatur posisi susunan bit pada control word nya.

Mode - mode tersebut adalah :

1. Mode 0
2. Mode 1
3. Mode 2

Pada bagian ini hanya dibahas mengenai mode 0, karena sesuai dengan perancangan alat yang digunakan.

#### MODE 0 ( BASIC I/O )

Untuk menciptakan I/O yang non conditional atau yang non handshaking maka dipakai mode 0, yang pengoperasiannya adalah memberikan control word terlebih dahulu pada control register. Control word tersebut akan menentukan register yang akan dipakai pada 8255. Penentuan bit pada control word untuk menentukan operating mode dapat dilihat pada mode Definition format pada gambar 2.9

Dalam gambar tersebut ketiga port terbagi dalam dua grup pilihan mode. Yaitu Grup A untuk port A dan port C upper yang dapat diprogram dari mode 0 sampai mode 2, sedangkan grup B untuk port B dan port C lower untuk pengoperasian mode 0 dan mode 1.

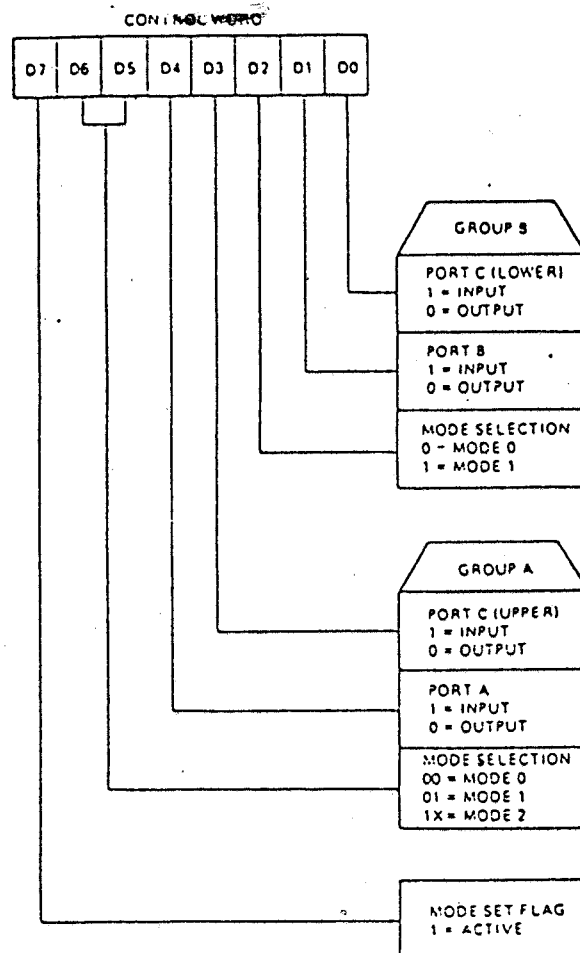
Control word terdiri dari delapan bit data yang dapat dinotasikan sebagai berikut :

D7	D6	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----

- D7 ( mode set flag ).  
Menentukan aktif dan tidaknya 8255 saat beroperasi.  
Aktif bila berlogika "1" berarti control word dikirimkan ke control register dan tidak aktif bila berlogika " 0 ".
- D6 dan D5 (Mode selection group A).  
Digunakan untuk menentukan mode operasi yang diinginkan pada port A.
- D4 ( port A )  
Menunjukkan fungsi port A sebagai input atau output.  
Bila logik "1" berarti sebagai input dan bila logik "0" berarti sebagai output.
- D3 (port C Upper)  
Menentukan port C Upper sebagai input atau output.  
Bila logic "1" berfungsi sebagai input dan bila logic "0" sebagai output.
- D2 ( mode selection Grup)  
Menentukan mode operasi grup B. pada mode 0 logic "0".
- D1 ( Port B ).  
Menentukan fungsi port sebagai input atau output.  
Bila logic "1" berfungsi sebagai input dan bila logic "0" berfungsi sebagai output.
- D0 ( port C lower ).  
Menentukan fungsi 4 bit port C lower sebagai input atau output. Bila berfungsi sebagai input logic "0"

dan bila sebagai output berlogik "1".

Dengan mengkombinasikan susunan logic dari control register tersebut PPI dapat memberikan 24 fungsi input/output yang dapat dihubungkan dengan peralatan luar.



Gambar 2.9.11

format mode operasi 8255

## II.15. HUBUNGAN 8088 DENGAN SLOT YANG DIPAKAI UNTUK I/O

### SLOT

Isi masing - masing slot pada IBM PC adalah :

1. 8 bit sistem data bus.
2. 20 pin address bus.
3. Sumber daya dan ground untuk rangkaian interface.
4. Clock dan timing signal.
5. Jalur control untuk memory I/O read dan Write.
6. Jalur maskable dan non maskable interrupt request.
7. Jalur status untuk interrupt enable Flag.
8. Jalur status bus cycle.

Jalur sinyal ini terdapat 62 pin pada slot yang jenis dan penggunaannya adalah sebagai berikut :

#### 1) D0 - D7

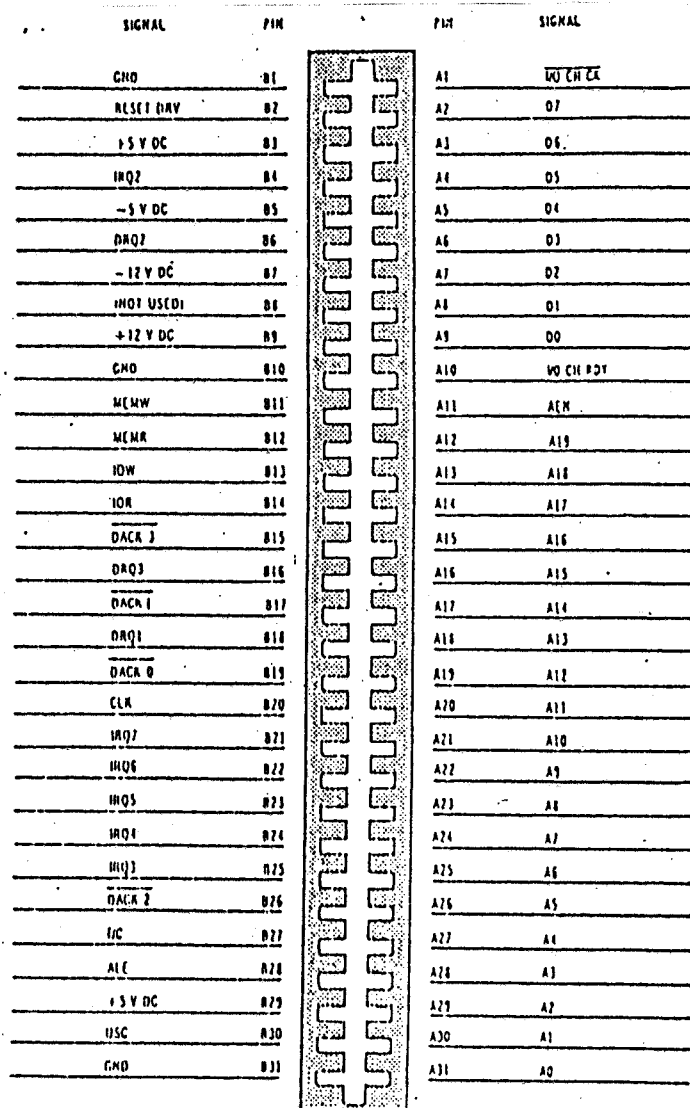
Jalur ini dihubungkan dengan sistem data bus yang digunakan sebagai jalur data input maupun output. Jalur ini adalah tristate dan aktif high.

#### 2) A0 - A19

Jalur ini dihubungkan dengan sistem address bus, yang akan memberi address pada memory dan peralatan I/O. Jalur ini merupakan jalur output yang bersifat tri - state yang berguna untuk memberi address pada decoder agar dihasilkan sinyal enable.

#### 3) ALE

Singkatan dari Address Latch Enable merupakan jalur output yang dibangkitkan oleh prosesor untuk melatch address yang sudah tepat benar (valid) pada proses



Gambar 2.9<sup>12</sup>

Gambar Slot IBM PC

demultipleks jalur alamat dan data. Sedangkan pada slot jalur data dan address sudah dipisahkan atau dilatch, maka ALE hanya untuk menunjukkan bahwa proses demultipleks sedang terjadi.

12). Lewis. C. Eggebrecht, Op cit, hal 77

latch, maka ALE hanya untuk menunjukkan bahwa proses demultipleks sedang terjadi.

4) DRQ

Singkatan dari Direct - memory access request yang merupakan sinyal input dari I/O untuk memberitahu DMA control bahwa unit I/O tersebut high ke jalur DRQ. Sedangkan DMA controller memberikan jawaban sinyal DACK (Direct memory access ACKnowledge) bahwa permintaan sinyal DRQ sudah diterima dan sistem bus sudah dalam kuasa DMA controller yang siap melakukan operasi DMA. Pada 8237 yang bekerja sebagai DMA controller memiliki DRQ dan DACK yang masing - masing 4 jalur yang diberi nomor 0,1,2 dan 3. Jalur tersebut memiliki prioritas untuk dilayani dan dapat diprogram. Didalam BIOS IBM PC telah berisi program yang menyebabkan DRQ0 memiliki prioritas paling tinggi dan DRQ3 paling rendah. Prioritas yang paling tinggi tersebut dari DRQ0 sudah dipakai dalam sistem board untuk keperluan refresh RAM dinamis sehingga jalur DACK0 tersedia pada slot dan dipakai untuk mengetahui bahwa sedang terjadi dummy read (refresh) pada memory dinamis.

5) OSC

Merupakan sinyal output dari operasi oscillator pada sistem board dengan frekuensi kristal 14,31818 Mhz yang dibangkitkan oleh generator clock 8284 dan juga menghasilkan 4,772727 Mhz untuk clock sistem. demikian juga untuk input clock 8253 PTC 1,1931817 Mhz dari 4,772727 Mhz yang akan dibagi dengan D flip - flop.



## 6) CLK

Adalah sinyal output pada I/O port dengan frekuensi 4,772727 Mhz seperti yang disebutkan diatas, bahwa IC 8284 bekerja untuk menghasilkan pulsa clock dan pulsa untuk sinkronisasi. Pertama kali jika komputer dinyalakan maka 8284 akan menerima sinyal POWER GOOD dari power supply, yang menyatakan power supply dalam keadaan baik. Dengan adanya sinyal POWER GOOD tersebut 8284 akan menghasilkan sinyal RESET ke 8088 yang mengakibatkan terjadinya inisialisasi storage register. Sedangkan 8284 untuk I/O port akan memberikan sinyal READY ke 8088, yaitu untuk menyatakan bahwa proses perpindahan data antara 8088 dan I/O port berjalan dengan baik, bila I/O port tidak bisa mengikuti kecepatan 8088, maka I/O port dapat mengirimkan sinyal kerangkaian penghasil pulsa tunggu yang akan diteruskan ke 8284. Dengan adanya sinyal tunggu pada 8284 maka 8284 akan memberikan informasi ke 8088 lewat sinyal ready agar 8088 menunda pelaksanaan tugasnya beberapa saat (selama sinyal tunggu dipancarkan) agar pelaksanaan pemindahan data berlangsung dengan sinkron.

## 7) IRQ2 s/d IRQ7

Merupakan jalur input yang berhubungan langsung dengan 8259 interrupt controller, sehingga dapat dipakai oleh unit I/O untuk menginterruptsi 8088 CPU. Pada 8259 memiliki 8 jalur interrupt tetapi hanya 6 jalur interrupt yang tersedia pada I/O port karena IRQ0 dan IRQ1 dipakai oleh sistem board.

## 8) IOR, IOW, MEMR, dan MEMW

Merupakan sinyal output yang digunakan untuk pengatur, sinyal IOR untuk I/O baca, sinyal IOW untuk I/O tulis, sinyal MEMR untuk Memory Read, sinyal MEMW untuk memory Write. Sinyal - sinyal tersebut dibangkitkan oleh CPU 8088 atau oleh 8237 DMA controller dengan aktif rendah.

## 9) TC

Merupakan sinyal aktif tinggi yang dikeluarkan oleh DMA controller. Sinyal tersebut dipakai untuk menunjukkan bahwa salah satu channel DMA telah mencapai suatu jumlah cycle transfer tertentu yang diprogram sebelumnya. Sinyal tersebut digunakan untuk menghentikan DMA yang mentransfer satu blok data. Karena sinyal tersebut dikeluarkan jika salah satu dari 4 channel DMA mencapai hitungan tertentu, maka untuk mengetahui channel mana yang dicapai TC dengan dipakainya sinyal DACK yang di AND dengan sinyal TC tersebut

## 10) RESET DRV

Merupakan sinyal output yang dikeluarkan oleh DMA controller untuk menunjukkan bahwa operasi DMA sedang terjadi, sehingga dipakai untuk I/O sebagai pencegahan terjadinya pengkodean I/O address pada saat terjadinya proses DMA, karena hal ini semua bus dalam penguasaan DMA controller.

## 12) I/O CH CK

Merupakan sinyal input yang aktif rendah. Sinyal ini dikeluarkan oleh interface pada slot agar dapat



melakukan NMI ke 8088 CPU. Biasanya dipakai oleh interface untuk menunjukkan bahwa telah terjadi kesalahan pada interface card.

#### 13) I/O CH RDY

Merupakan sinyal input yang digunakan untuk menunjukkan kesiapan dari peralatan I/O pada saat berkomunikasi dengan CPU. dengan sinyal tersebut dapat dihasilkan tambahan sinyal tunggu pada bus cycle.

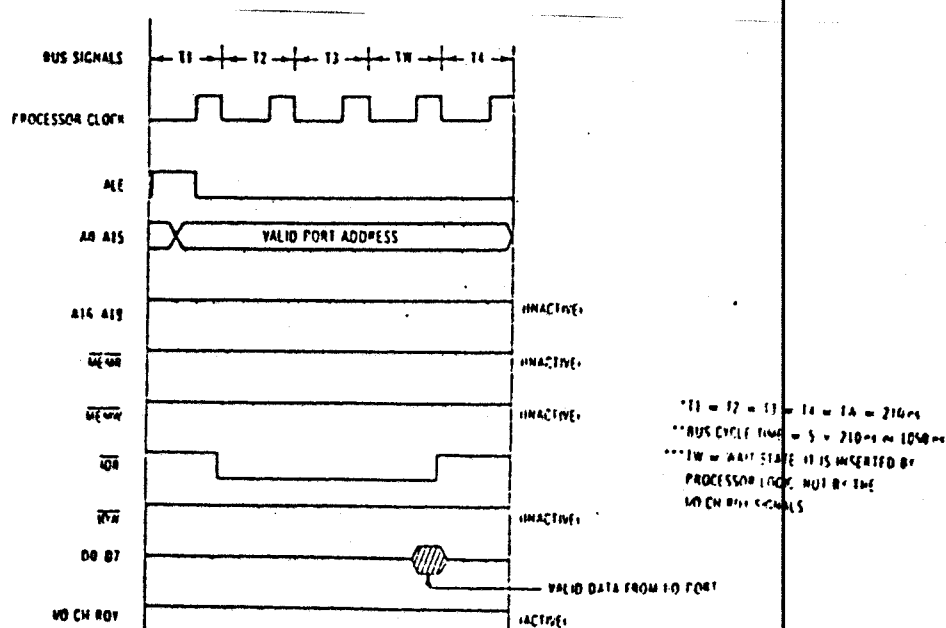
#### 14) Power Supply dan Ground

Merupakan penyedia sumber daya untuk unit interface. pada slot tersedia +5, -5, +12 dan -12 Volt terhadap ground. daya yang tersedia oleh power supply umumnya sebesar 63 watt.

### II.15 SIKLUS BACA I/O PORT

Siklus ini diaktifkan setiap kali mikroprosessor 8088 menemui instruksi IN. Siklus ini berfungsi untuk mengambil data dari salah satu I/O port pada lokasi address I/O port. Dalam aplikasi PC, siklus tersebut memerlukan minimum lima pulsa clock atau satu microdetik. Siklus ini dapat diperpanjang dengan menonaktifkan pin ready pada mikroprosessor 8088. Hal ini terjadi untuk peralatan I/O yang lambat. Ketika terjadi siklus baca I/O port ini, mikroprosessor 8088 mengirimkan address 16 bit. ( A0 - A15 ) ke sistem address bus. Pada siklus tersebut empat bit tertinggi ( A16 - A19 ) tidak aktif. Timing diagram dari siklus baca I/O tersebut dapat digambarkan pada gambar 2. berikut. Saat periode T1, sinyal ALE diaktifkan untuk menunjukkan bahwa address A0 - A15 mengandung address I/O

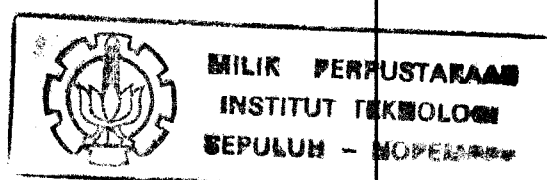
Port yang valid. Pada periode T2, sinyal dari bus control IOR diaktifkan yang menunjukkan bahwa siklus bus adalah baca I/O port dan mengakibatkan port mengirim data ke data bus. Pada awal sinyal clock T4, prosessor mengambil data dari data bus dan sinyal bus IOR di nonaktifkan. Maka siklus T4 adalah akhir dari siklus bus keseluruhan. Di dalam siklus baca I/O terdapat empat siklus sedangkan penggunaan untuk PC disisipkan satu siklus TW secara otomatis dalam setiap siklusnya.



Gambar 2.10<sup>13)</sup>

Siklus Baca I/O port

13) Eggebrecht, C. Lewis, op cit hal 47



## II.17 SIKLUS TULIS I/O PORT

Siklus ini dimulai saat mikroprosesor menjalankan perintah out. Siklus menulis data dari mikroprosesor 8088 ke suatu address yang ditentukan pada range address I/O port, Seperti juga siklus baca I/O port, pada siklus ini mempunyai empat siklus clock dan dalam PC secara otomatis akan disisipi oleh sinyal TW. Jadi di dalam PC seluruh siklus tulis I/O port terdiri dari lima sinyal clock atau sekitar 1.05 mikrodetik. Siklus tersebut dapat diperpanjang dengan menggunakan sinyal READY. Selama siklus ini address bus yang digunakan adalah A0 - A15 sedangkan address A16 - A19 tidak aktif. Sama dengan siklus - siklus lain sinyal ALE diaktifkan selama periode T1 dan hal ini menunjukkan address bus address bus berisi address port yang valid. Pada saat T2 sinyal bus control IOW akan aktif, hal ini menunjukkan siklus bus adalah siklus tulis I/O dan akan membawa data dari data bus. Setelah siklus T2 mikroprosesor akan mengirim data ke port address. Pada awal siklus T4, sinyal control IOW adalah tidak aktif maka siklus akan berakhir dengan selesainya sinyal T4



## II.18 BUFFER ( PENYANGGA ).

Dalam mengendalikan IC - IC atau I/O Device dari suatu sistem mikroprosesor, maka mikroprosesor memerlukan buffer untuk menambah kapasitas penyediaan arus pin - pin busnya. Buffer sangat diperlukan karena jika mikroprosesor berlebihan ( fan outnya dilampaui), akibatnya level tegangan pada pin - pin bus yang bersangkutan dapat turun sampai melampaui batas marginnya yang mengakibatkan mikroprosesor memberikan informasi yang tidak benar pada seluruh sistem. Untuk menghindari hal tersebut maka digunakan buffer pada sistem busnya.

Ada bermacam - macam buffer, ada yang inverting, non inverting, maupun yang tri state buffer. Tri state buffer adalah type buffer yang sering digunakan, kelebihanannya adalah bila buffer ini tidak digunakan maka kondisi dari buffer tersebut dalam keadaan high impedance yaitu tidak membebani komponen lainnya.

Dalam penggunaannya harus juga diketahui jenis dari buffer yang akan digunakan. Untuk bus yang unidirectional ( address bus dan kontrol bus) dapat digunakan IC 74LS244 (octal bus driver), sedangkan untuk bus yang bidirectional dapat dipakai ( octal bus transceiver).

## II.19 PENGUBAH ANALOG KE DIGITAL

Pengubah analog ke digital digunakan untuk mengubah keadaan dari suatu keadaan analog menjadi keadaan digital dalam hal ini terdiri dari dua keadaan "On" dan "Off". Resolusi dari pengubah Analog ke digital didasarkan pada jumlah bit pada output biner.

Spesifikasi dari pengubah analog ke Digital didasarkan pada "conversion time", disini akan dibahas macam - macam pengubah analog ke digital berdasarkan conversion timenya :

### II.19.1. PARALLEL COMPARATOR A/D CONVERTER

Sebuah pembagi tegangan akan mengatur tegangan referensi pada input inverting dari masing - masing comparator. Tegangan pada sisi puncak dari rangkaian pembagi akan menampilkan nilai penuh dari pengubah. Tegangan yang dikonversikan diterapkan pada non inverting input dari semua comparator yang bekerja paralel. Jika tegangan input dari comparator lebih besar daripada tegangan referensi pada input inverting maka output comparator akan high. Output dari comparator akan memberikan tampilan digital dari level tegangan input.

Keuntungan penggunaan parallel A/D converter adalah kecepatannya dalam mengkonversi, yang menyederhanakan propagasi delay time dari comparator. Sedangkan kerugian penggunaan pengubah ini adalah banyaknya comparator yang digunakan untuk mendapatkan output dengan resolusi yang baik.

### II.19.2 DUAL SLOPE A/D CONVERTER

jenis pengubah ini banyak digunakan pada peralatan pengukur digital karena dapat memberikan resolusi bit yang besar dengan biaya yang murah,

cara kerja converter ini ialah : dengan menstart maka control rangkaian akan mereset counter ke posisi nol kembali dan akan menghubungkan input integrator dengan input tegangan yang akan dikonversikan. Jika input tegangan adalah positif maka outputnya adalah gelombang berbentuk ramp negatif. Kelemahan dari converter jenis ini adalah lambatnya kecepatan konversi yang diberikan.

### II.19.3 SUCCESSIVE APPROXIMATION A/D CONVERTERS

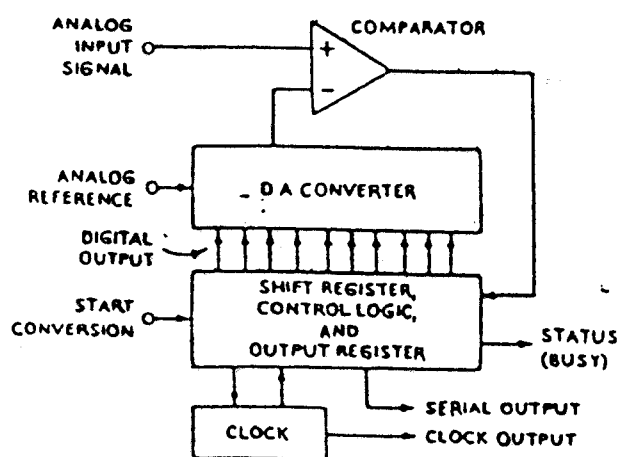
Successive approximation converter adalah converter yang tercepat karena memproduksi hanya beberapa kali delay konversi. Konversi dimulai dengan membuat inputnya menjadi nol. Pengubah jenis ini tidak menggunakan pencacah untuk memberikan masukan ke blok pengubah digital ke analognya. Sebagai gantinya, pengubah menggunakan register pendekatan berurutan (SAR).

Pada metode ramp integrating, pencacah yang digunakan akan maju melalui tiap - tiap hitungan biner langkah demi langkah. Sedangkan pada SA - ADC, register mempunyai harga - harga biner yang dimodifikasikan dalam urutan - urutan yang jauh berbeda. Urutan tersebut adalah sebagai berikut :

- MSB diset ke logika '1' dan bit - bit lainnya diisi dengan logika '0'. Hal ini menghasilkan harga Vout pada keluaran DAC sama dengan bobot dari MSB. Apabila Vout lebih besar daripada Vin, maka keluaran pembanding akan berubah ke taraf rendah. Hal ini akan mereset MSB. Apabila Vout lebih kecil daripada Vin, MSB tetap pada logika '1'.

- MSB kedua akan diset ke '1', apabila harga baru dari Vout lebih besar daripada Vin, bit ini akan direset ke '0'. Bila lebih kecil harganya tetap '1'.

- Proses ini berlangsung terus untuk seluruh bit dalam registernya. Proses pendekatan ini memerlukan satu periode detak untuk setiap bit. Hal ini merupakan kelebihan dari SA - ADC. Jadi

Gambar 2.12<sup>15)</sup>

Blok Diagram SA - ADC

15) Analog to Digital Conversion Handbook, Prentice Hall  
Englewood Cliffs, New York 1986 hal 213



bila menggunakan SA - ADC dengan resolusi 8 - bit, dibutuhkan 8 pulsa detak untuk menyelesaikan setiap konversi.

#### II.19.4. ADC 0808

IC ADC 0808 adalah komponen data acquisition yang terbuat dengan teknologi CHMOS, mempunyai 8 saluran input yang dapat dipilih dengan memberikan logika tertentu pada pin - pin inputnya.

##### II.19.4.1. BLOK DIAGRAM ADC 0808

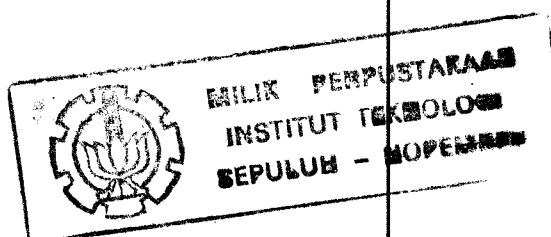
Gambar 2. menunjukkan blok diagram dari ADC 0808 dimana terdapat beberapa jalur input maupun output.

ADDRES LINE			ANALOG CHANNEL TERPILIH
C	B	A	
L	L	L	IN : 0
L	L	H	IN : 1
L	H	L	IN : 2
L	H	H	IN : 3
H	L	L	IN : 4
H	L	H	IN : 5
H	H	L	IN : 6
H	H	H	IN : 7

Tabel 2.4 16

Pemilihan saluran input

16) National Semiconductor Corporation  
Linear Data book, National Corp, 1982, hal 8 - 63



1. IN0 - IN7 (input)

Merupakan pin input untuk sinyal analog yang akan diubah menjadi bentuk logika biner. Sinyal input ini harus berharga positif dan maksimum adalah 5 volt.

2.  $2^{-1}$  -  $2^{-8}$  (Output)

Merupakan pin output yang berbentuk logika '1' dan '0'. Dimana kombinasi kedelapan logika ini menunjukkan besarnya tegangan sinyal analog yang terkonversi.  $2^{-1}$  merupakan most significant bit sedangkan  $2^{-8}$  merupakan least significant bit.

3. ADD A - C (Decoder)

Pin ini merupakan pin input dimana kombinasi logika ketiga dari pin - pin ini akan memilih salah satu input dari kedelapan sinyal input yang akan diubah.

4. Clock

Merupakan input clock untuk ADC ini, dimana frekwensi untuk clock ini berharga 10 - 1200 Khz.

5. Start

Merupakan pin input yang aktif pada saat perobahan dari logika low menjadi logika high. Bila sinyal ini aktif maka ADC ini mula mengubah sinyal input.

6. ALE (Address input Enable)

Merupakan pin input yang aktif high, sinyal pada input ini menunjukkan bahwa ADC akan mengunci

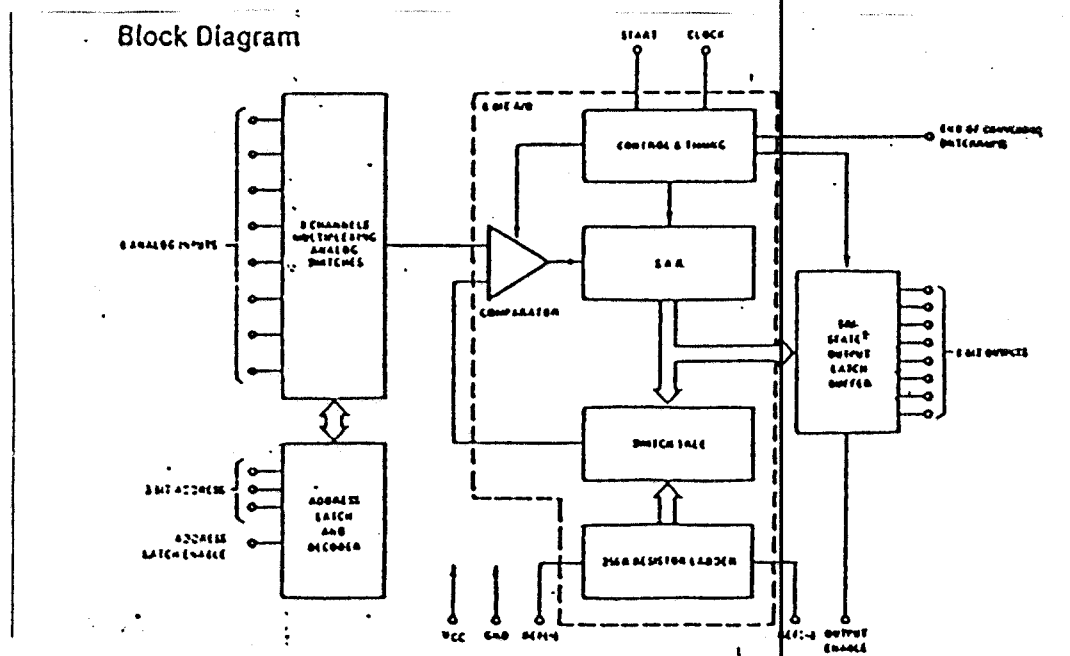
sinyal logika pada decoder supaya tidak berubah selama operasi konversi.

7. EOC (End of Conversion)

Merupakan pin output yang aktif high dimana selama ADC masih dalam proses konversi maka output pada EOC akan berlogika low dan bila proses konversi selesai, Output pin ini akan berlogika high.

8. OE (output Enable)

Merupakan pin input yang aktif high dimana setelah sinyal OE high dan sinyal OE ini juga aktif maka data hasil konversi pada ADC akan ditempatkan pada data bus untuk selanjutnya siap dibaca oleh mikroprosessor.



Gambar 2.13 <sup>17)</sup>

BLOK DIAGRAM 0808

17) ibid hal 8 - 60

9. + REF (Referensi positif)

Merupakan sinyal input yang menunjukkan harga maksimum untuk sinyal analog yang akan dikonversikan. Untuk ADC 0808 ini, harga yang diijinkan adalah = Vcc

10. - REF ( Referensi negatif)

Tegangan yang dikenakan pada pin - pin ini menunjukkan harga terendah yang diijinkan untuk input analog.

11. Vcc ( Suplly +)

12. Gnd ( Ground ).

## BAB III PERENCANAAN

### III.1. PENDAHULUAN

Rangkaian pembantu untuk synchronisasi generator yang direncanakan ini terdiri dari tiga modul, yaitu :

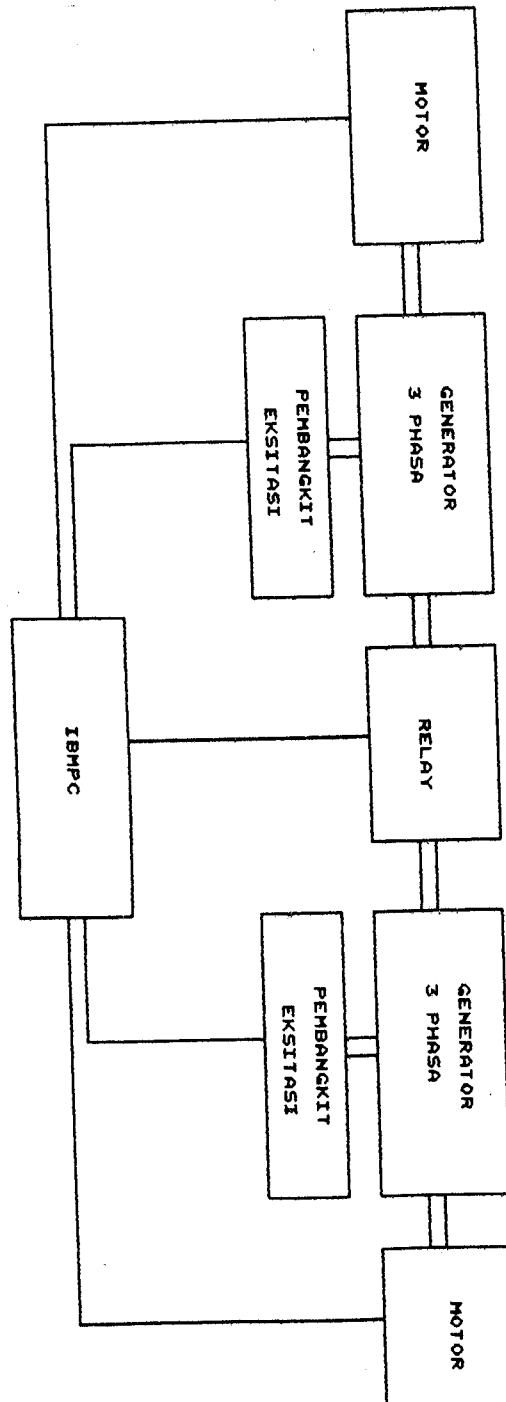
1. Modul Pengatur Utama
2. Modul Pendeteksi besaran analog
3. Modul Pengatur putaran motor

Modul pengatur utama dalam hal ini adalah komputer IBM PC - XT yang digunakan untuk mengatur keseluruhan proses yang sedang terjadi dalam peralatan. Fungsi dari pendeteksi besaran analog adalah untuk mendeteksi semua persyaratan untuk melakukan kerja paralel generator. Peralatan ini mendeteksi tegangan, frekwensi dan fasa dari generator. Sedangkan fungsi dari rangkaian pengatur motor adalah untuk mengatur putaran motor ke kiri dan ke kanan untuk mensinkronkan syarat - syarat yang belum terpenuhi.

### III.2 BLOK DIAGRAM

Perencanaan dan pembuatan alat untuk synchronisasi generator tiga phasa yang diinterfacekan ke IBM PC XT adalah sebagai berikut :

- Rangkaian Interface yang menghubungkan modul driver dengan IBM PC XT.
- Rangkaian analog untuk pengukuran parameter - parameter



Gambar 3.1

Blok Diagram Peralatan yang direncanakan

yang diperlukan untuk syarat kerja paralel, seperti :  
tegangan, frekwensi dan urutan fasa dari generator

- Rangkaian control motor yang dalam hal ini motor digunakan untuk prime mover dari generator yang akan dioperasikan paralel.
- Rangkaian untuk mengatur pemberian exitasi kepada motor dan generator yang dalam hal ini diberikan oleh sebuah sumber DC yang dapat di atur.

Blok diagram rangkaian yang direncanakan adalah sebagai yang digambarkan pada gambar 3.1

Dalam gambar tersebut terlihat bahwa besaran analog yang di dapatkan akan dibaca oleh CPU dan dibandingkan antara ke duanya

### III.3 RANGKAIAN INTERFACE

Untuk perancangan interface IBM yang perlu diperhatikan adalah :

- Address memory mapping dan decoding pada pada IBM PC - XT.
- I/O port mapping dan decoding pada IBM PC - XT.
- Rangkaian untuk bus request.
- Rangkaian untuk bus driver dan receiver card untuk IBM PC

#### 111.4. I/O PORT MAPPING DAN DECODING PADA IBM PC XT

Arsitektur 8088 sebenarnya memungkinkan pengadressan I/O port sampai sebanyak 65,536 lokasi. Tetapi dalam perencanaan IBM PC XT, tidak seluruh lokasi tersebut dapat diakses. Hanya 10 bit terendah dari address bus (A0 - A9) yang digunakan untuk address decoding bagi peralatan I/O.

Dalam perencanaan IBM PC XT, bit A9 dari I/O port address bus memiliki arti khusus. Jika bit ini tidak aktif (logika low), data hanya dapat diterima dari peralatan yang terdapat pada sistem board. Tetapi, jika bit ini aktif (berlogika high) data dapat diterima dari peralatan I/O yang terpasang pada salah satu dari kedelapan expansion slot yang ada pada sistem board. Jadi 1024 lokasi I/O port yang tersedia pada IBM PC XT terbagi atas dua bagian, yaitu 512 lokasi yang terdapat pada sistem board dan 512 lokasi yang terdapat pada sistem board dan 512 lokasi terdapat pada card slot bus. Pembagian ini hanya berlaku untuk input port, dan tidak berlaku untuk output port.

Penggunaan lokasi memory pada pada alamat 0200H - 03FFH terdapat pada ekspansion slot pada IBM dimana tidak seluruh pengalamatan dapat digunakan karena sudah dipakai untuk peralatan tambahan berupa : Monochrom / Color Graphic Adapter, Printer port Adapter, Disket Drive Adapter dan seterusnya. Penggunaan alamat tersebut dapat dilihat pada gambar 3.





HEX ADDRESS		USES
0200H	1	NOT USED
0201H	1	GAME CONTROL ADAPTER
0202H	118	NOT USED
0277H	8	NOT USED
0278H	8	SECOND PRINTER PORT ADAPTER
0279H	8	NOT USED
027AH	8	NOT USED
027BH	8	SECOND SERIAL PORT ADAPTER CARD
027CH	8	NOT USED
027DH	8	NOT USED
027EH	8	NOT USED
027FH	8	NOT USED
0280H	170	NOT USED
0281H	170	NOT USED
0282H	170	NOT USED
0283H	170	NOT USED
0284H	170	NOT USED
0285H	170	NOT USED
0286H	170	NOT USED
0287H	170	NOT USED
0288H	170	NOT USED
0289H	170	NOT USED
028AH	170	NOT USED
028BH	170	NOT USED
028CH	170	NOT USED
028DH	170	NOT USED
028EH	170	NOT USED
028FH	170	NOT USED
0290H	16	MONOCHROME AND PRINTER ADAPTER
0291H	16	NOT USED
0292H	16	NOT USED
0293H	16	NOT USED
0294H	16	NOT USED
0295H	16	NOT USED
0296H	16	NOT USED
0297H	16	NOT USED
0298H	16	NOT USED
0299H	16	NOT USED
029AH	16	NOT USED
029BH	16	NOT USED
029CH	16	NOT USED
029DH	16	NOT USED
029EH	16	NOT USED
029FH	16	NOT USED
02A0H	16	NOT USED
02A1H	16	NOT USED
02A2H	16	NOT USED
02A3H	16	NOT USED
02A4H	16	NOT USED
02A5H	16	NOT USED
02A6H	16	NOT USED
02A7H	16	NOT USED
02A8H	16	NOT USED
02A9H	16	NOT USED
02AAH	16	NOT USED
02ABH	16	NOT USED
02ACH	16	NOT USED
02ADH	16	NOT USED
02AEH	16	NOT USED
02AFH	16	NOT USED
02B0H	16	NOT USED
02B1H	16	NOT USED
02B2H	16	NOT USED
02B3H	16	NOT USED
02B4H	16	NOT USED
02B5H	16	NOT USED
02B6H	16	NOT USED
02B7H	16	NOT USED
02B8H	16	NOT USED
02B9H	16	NOT USED
02BAH	16	NOT USED
02BBH	16	NOT USED
02BCH	16	NOT USED
02BDH	16	NOT USED
02BEH	16	NOT USED
02BFH	16	NOT USED
02C0H	16	NOT USED
02C1H	16	NOT USED
02C2H	16	NOT USED
02C3H	16	NOT USED
02C4H	16	NOT USED
02C5H	16	NOT USED
02C6H	16	NOT USED
02C7H	16	NOT USED
02C8H	16	NOT USED
02C9H	16	NOT USED
02CAH	16	NOT USED
02CBH	16	NOT USED
02CH	16	NOT USED
02D0H	16	NOT USED
02D1H	16	NOT USED
02D2H	16	NOT USED
02D3H	16	NOT USED
02D4H	16	NOT USED
02D5H	16	NOT USED
02D6H	16	NOT USED
02D7H	16	NOT USED
02D8H	16	NOT USED
02D9H	16	NOT USED
02DAH	16	NOT USED
02DBH	16	NOT USED
02DCH	16	NOT USED
02DDH	16	NOT USED
02DEH	16	NOT USED
02DFH	16	NOT USED
02E0H	16	NOT USED
02E1H	16	NOT USED
02E2H	16	NOT USED
02E3H	16	NOT USED
02E4H	16	NOT USED
02E5H	16	NOT USED
02E6H	16	NOT USED
02E7H	16	NOT USED
02E8H	16	NOT USED
02E9H	16	NOT USED
02EAH	16	NOT USED
02EBH	16	NOT USED
02ECH	16	NOT USED
02EDH	16	NOT USED
02EEH	16	NOT USED
02EFH	16	NOT USED
02F0H	16	NOT USED
02F1H	16	NOT USED
02F2H	16	NOT USED
02F3H	16	NOT USED
02F4H	16	NOT USED
02F5H	16	NOT USED
02F6H	16	NOT USED
02F7H	16	NOT USED
02F8H	16	NOT USED
02F9H	16	NOT USED
02FAH	16	NOT USED
02FBH	16	NOT USED
02FCH	16	NOT USED
02FDH	16	NOT USED
02FEH	16	NOT USED
02FFH	16	NOT USED

NOTE: NEW FEATURES BY IBM AND OTHER MANUFACTURERS MAY USE SOME OF THE SPARE I/O ADDRESS DECODES

Gambar 3.3

Penggunaan alamat I/O pada slot IBM

### 111.5 PERENCANAAN ADDRESS DECODING I/O PORT ADDRESS DECODING

Melihat alokasi alamat yang kosong diatas maka alamat untuk I/O peralatan ini menggunakan alamat 0200H - 0203H untuk PPI satu dan 0300H - 0303H untuk PPI dua , jadi

Untuk PPI satu

Port A : address 0200H

Port B : address 0201H

Port C : address 0202H

Control Word : address 0203H

Untuk PPI dua

Port A : address 0300H

Port B : address 0301H

Port C : address 0302H

Control Word : address 0303H

maka untuk PPI satu mempunyai tabel kebenaran sebagai berikut

Tabel 3.1 Tabel kebenaran I/O Address decoding PPI satu

AEN	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	0	1	1

Tabel 3.2 Tabel kebenaran I/O Address decoding PPI dua

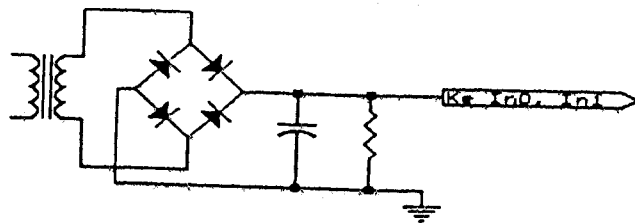
AEN	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	1	1	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	0	0	0	1
0	1	1	0	0	0	0	0	0	1	0
0	1	1	0	0	0	0	0	0	1	1

Sinyal AEN dari ekspansi slot IBM PC XT adalah aktif high yang dibangkitkan oleh rangkaian logika DMA Controller. Sinyal ini menunjukkan bahwa suatu DMA bus cycle sedang berlangsung. Pada sistem bus dari IBM PC XT, Sinyal ini digunakan untuk mendisable I/O port selama berlangsungnya siklus DMA. Hal ini dapat terjadi karena IOR dan IOW bisa aktif bersama address memory pada address bus selama berlangsungnya siklus DMA. Karena itu jalur sinyal AEN yang aktif high ini dihubungkan pada input G2A yang aktif low dari IC decoder 74LS138.

Rangkaian dekoder yang digunakan menggunakan IC Demultiplekser 74LS138. Untuk mengaktifkan 74LS138 diperlukan logika high pada pin 6 (G1) dan logika low pada pin 4 dan 5 (G2A dan G2B). Sedangkan untuk memilih saluran melalui pin 1, 2 dan 3. Kondisi pengaktifan IC 74LS138 dibentuk dari gerbang logika NOR kolektor terbuka untuk penghematan gerbang. Pemilihan alamat digunakan alamat I/O yang belum terisi sehingga dapat langsung di interfacekan dengan IBM PC XT. PPI satu dalam rangkaian ini digunakan untuk mengambil data dari ADC sedangkan untuk PPI dua digunakan untuk mengatur putaran motor dan switch. Maka untuk PPI satu rangkaian decodernya adalah sebagai berikut :



AC ke DC dimana dipasang empat buah diode dengan kombinasi jembatan dan menyearahkan gelombang penuh sinyal AC. Pada akhir rangkaian dipasang filter yang terdiri R dan C. Output dari rangkaian ini dimasukkan langsung kepada ADC input 0 dan 1. Sinyal yang diterima ADC harus merupakan sinyal positif.



Gambar 3.6

Rangkaian sensor tegangan

### III.7 UNIT RANGKAIAN SENSOR FREKWENSI

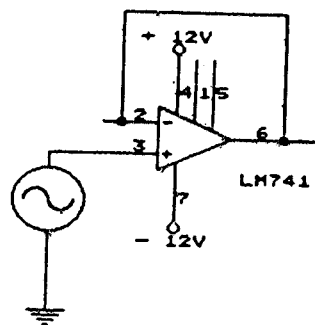
Unit Sensor Frekwensi terdiri dari :

1. Rangkaian Buffer
2. Rangkaian Pengubah Gelombang Sinus ke Persegi
3. Rangkaian Monostable

#### III.7.1. RANGKAIAN BUFFER ( PENYANGGA ).

Rangkaian Buffer (Penyangga) adalah rangkaian terdepan dari seluruh rangkaian ini yang berfungsi sebagai penyangga (Buffer) dan juga berfungsi sebagai rangkaian penyesuai impedansi bagi rangkaian berikutnya. Impedansi rangkaian buffer ini diharuskan besar (mendekati tak terhingga) agar

tidak membebani rangkaian berikutnya. Penyesuaian impedansi diperlukan untuk menghilangkan sebagian kesulitan tegangan offset masukan dan suseptibilitas derau. Rangkaian ini juga digunakan untuk mengamankan rangkaian berikutnya dari gangguan tegangan dari jala - jala. Menggunakan IC Op - Amp 741.



Gambar 3.4

#### Rangkaian Penyangga

Besarnya Gain penguatan tegangan sama dengan satu sehingga amplitudo tegangan output sama dengan amplitudo tegangan input. Rangkaian yang dipakai adalah rangkaian penguat non inverting maka bentuk sinyal output yang dihasilkan sama dan sefasa dengan sinyal inputnya.

#### 3.7.22. RANGKAIAN PENGUBAH GELOMBANG SINUS KE PERSEGI

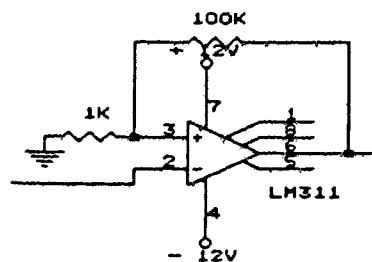
Fungsi dari rangkaian ini adalah mengubah gelombang sinus ke gelombang persegi dengan frekwensi output sama dengan frekwensi input serta amplitudo output yang konstan tidak terpengaruh oleh perubahan amplitudo input.

Gambar 3.3 menyajikan rangkaian pengubah gelombang sinus ke gelombang persegi. Pada dasarnya rangkaian tersebut adalah

merupakan rangkaian comparator dengan feedback positif.

Tujuan dipilihnya rangkaian ini adalah untuk menghindari efek gangguan dari sinyal pengganggu yang ada pada sinyal input.

Output dari rangkaian ini menjadi input Schmit Trigger untuk membuat lebih persegi (rata) output. Output dari rangkaian ini di inputkan ke sebuah Monostabil Multivibrator yang akan mengatur pewaktuan pulsa output. Digunakannya rangkaian Schmit Trigger adalah untuk menghilangkan sinyal tambahan (spike) yang terjadi pada saat berpindahnya dari level "Low" ke level "high" dalam hal ini supaya di dapatkan sinyal yang benar - benar persegi. Tegangan Output saat low adalah sebesar 0 volt karena pada saat low dioda menyumbat dan tahanan 10 K ohm menarik tegangan output menjadi 0 volt.



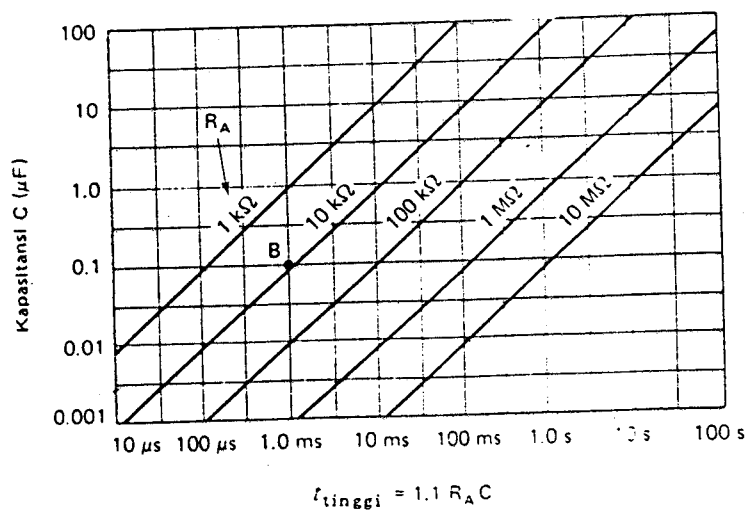
Gambar 3.3

Rangkaian Pengubah Gelombang Sinus  
Ke Gelombang Persegi



MILIK PERPUSTAKAAN  
INSTITUT TEKNOLOGI  
SEPULUH - NOPEMBER

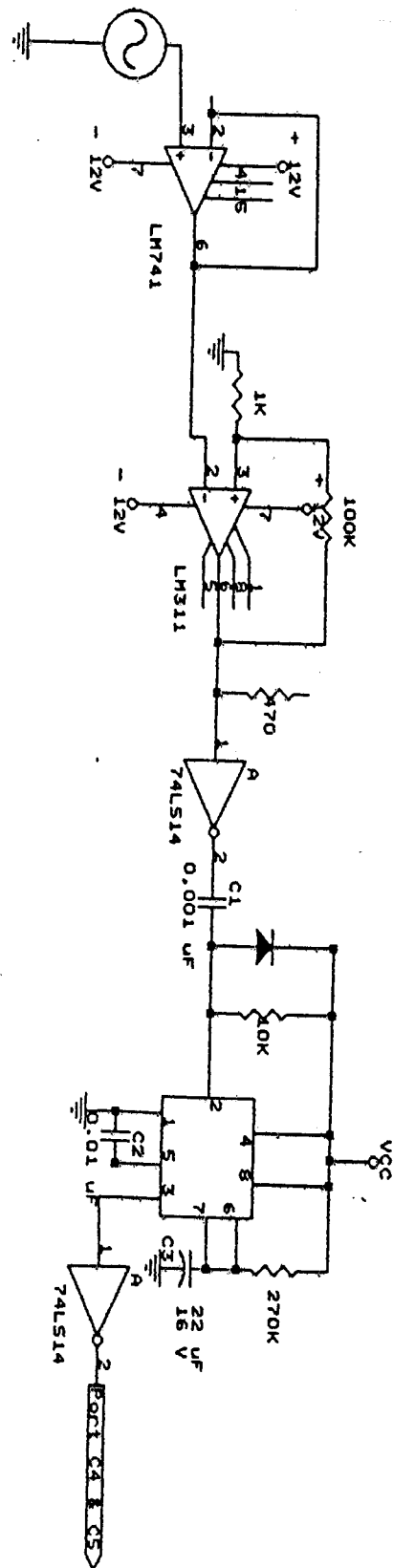




Gambar 3.4 17)

Grafik yang menentukan waktu denyut keluaran

Terlihat dalam gambar bahwa tahanan  $R_A$  dan Capacitor  $C$  menentukan waktu ketika sinyal outputnya dalam posisi high pada multivibrator. Perhitungan didasarkan kepada banyaknya sinyal yang terjadi dalam tiap detiknya.

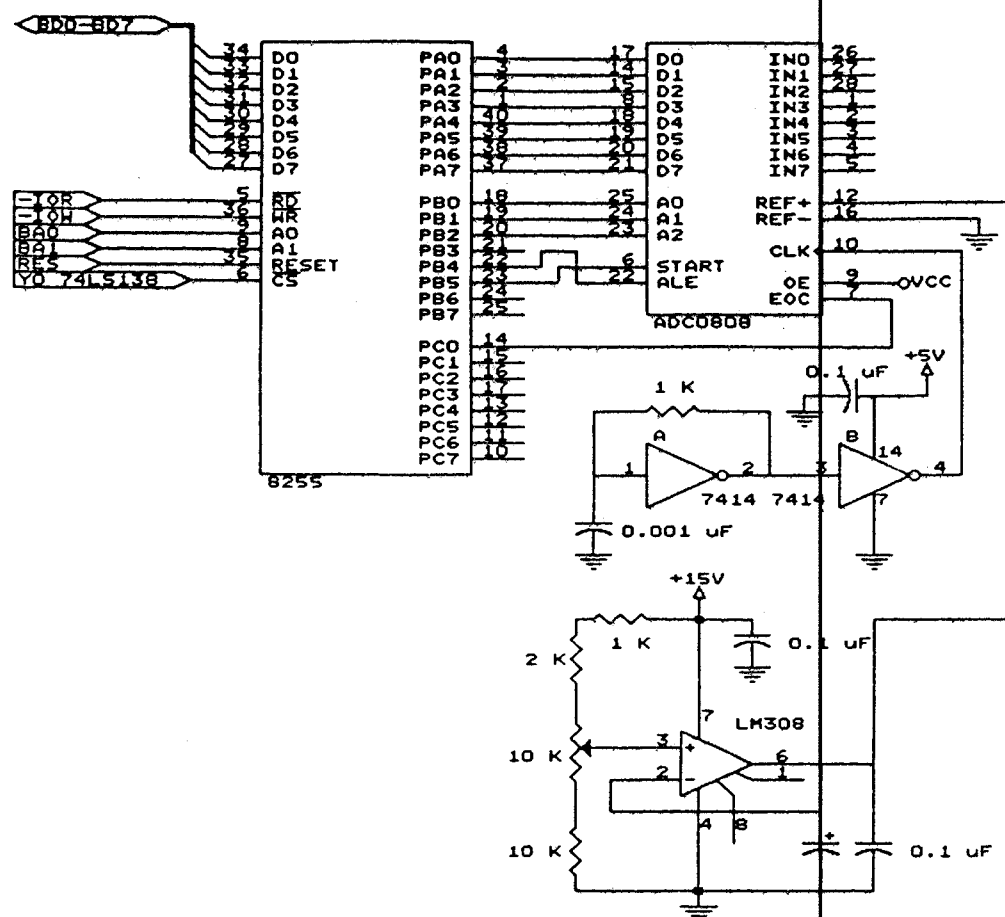


Gambar 3.5

Rangkaian lengkap sensor frekwensi

### 111.8 RANGKAIAN ADC

Rangkaian ADC menggunakan ADC 0808 sebagai komponen utamanya dimana rangkaian ini membutuhkan clock sinyal input sebesar 500 khz dimana sinyal clock sebesar itu di dapatkan dari kombinasi rangkaian R dan C yang dihubungkan dengan IC 741s14. Tegangan referensi positif 5 V didapatkan dari sebuah Opamp LM 308 sebagai pengikut tegangan dengan pengaturan kombinasi R variabel 10 K ohm. Rangkaian analog



Gambar 3.  
Rangkaian ADC

dijadikan input masing - masing input ADC dalam hal ini untuk input 1 dihubungkan dengan rangkaian sensor tegangan sedang yang lain dihubungkan dengan sensor frekwensi, sedangkan pin data langsung dihubungkan dengan Port A pada PPI dan pin Address dihubungkan dengan Port B dari PPI.

Pengoperasian ADC ini adalah sebagai berikut :

- Pada waktu Konversi :

Sinyal ADC ini akan aktif low dan sinyal WR juga aktif low Input ALE dan START aktif high maka ADC mulai melakukan konversi. selama konversi belum selesai EOC akan terus low dan jika konversi telah selesai maka EOC akan aktif high output EOC ini dihubungkan dengan PCO PPI.

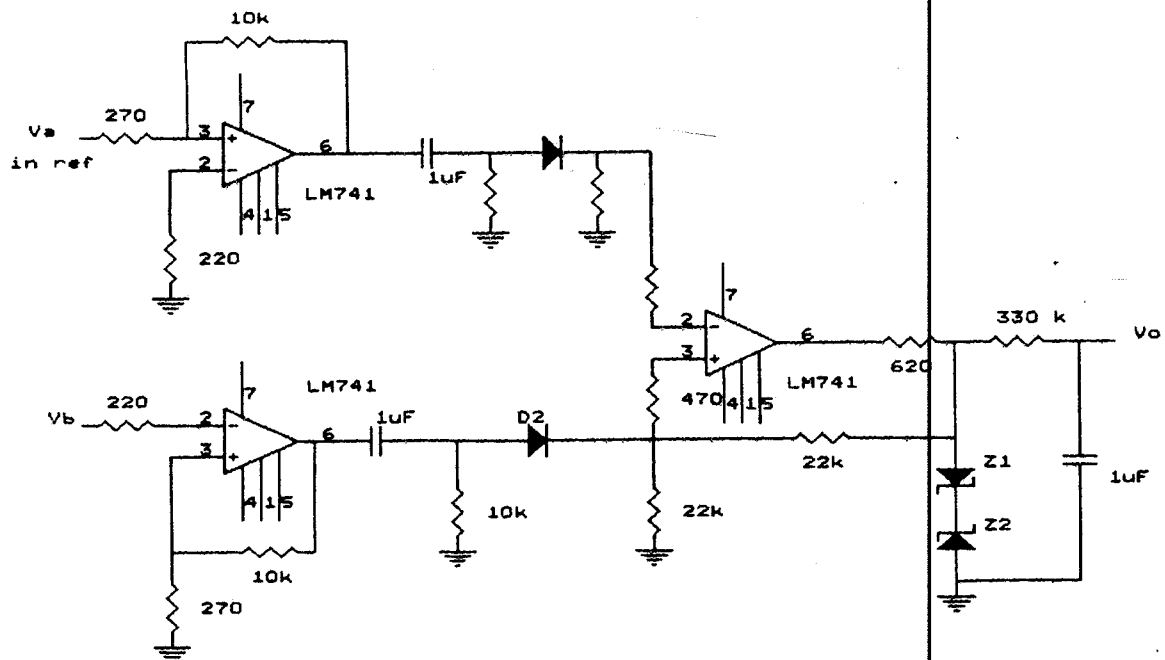
- Pengambilan Data

Pada waktu pengambilan data dilakukan maka CS dari ADC ini akan aktif low dan sinyal RD juga aktif low maka akan mengaktifkan OE karena disambungkan dengan Vcc maka akan high terus.

### 111.9 DETEKTOR PHASA

Rangkaian detektor phasa mengukur perbedaan phasa dari dua sinyal frekwensi yang sama. Seperti yang diperlihatkan dalam gambar

Pengukuran phasa yang dilakukan oleh rangkaian tersebut dapat terukur walaupun amplitudonya berbeda. Rangkaian tersebut menggunakan IC 741 yang dirangkai sebagai rangkaian



Gambar 3.6 .

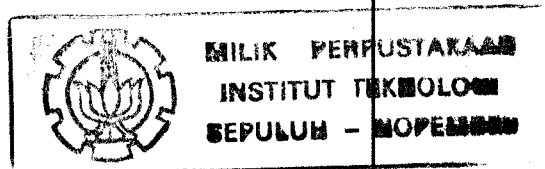
## Rangkaian Detektor Fasa

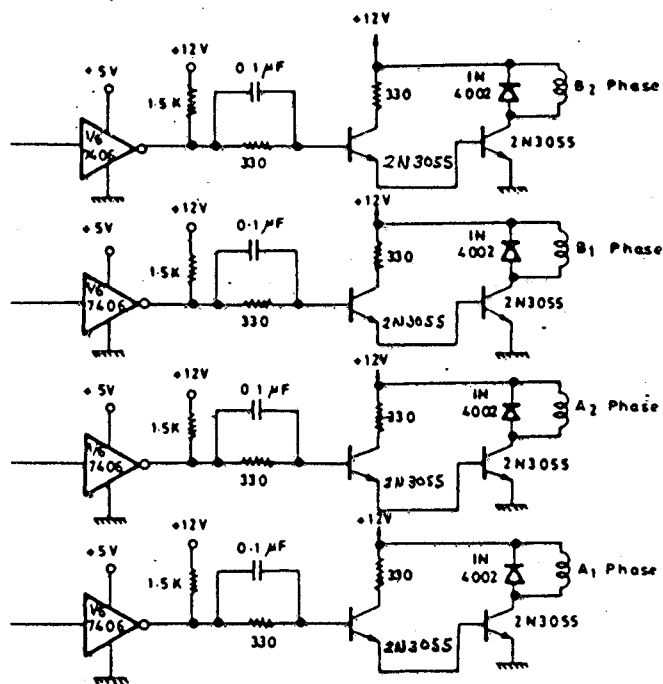
detektor penyilang nol. Disini OpAmp A2 dipasang sebagai rangkaian non inverting sedangkan A1 sebagai inverting. Kapasitor  $C1 = 1\mu F$  dan  $R7 = 10 K$  dan  $C2 = 1 \mu F$  dan  $R8 = 10K$  adalah berfungsi sebagai rangkaian deferensial, dimana

output dari rangkaian akan dipilih yang positif saja oleh diode D1 dan D2. Dan Diode Ditersebut akan membuat diode zener pada posisi low ( $V_5 = V_{Z1}$ ). Sedangkan pulsa D2 menyebabkan tegangan pada rangkaian diode zener berada pada posisi high ( $V_5 = V_{Z2}$ ). Jika  $V_a$  sephasa dengan  $V_b$  maka tegangan output sama dengan nol karena waktu yang dibutuhkan untuk keadaan high dan low adalah sama. Sedangkan bila  $V_b$  memimpin  $V_a$  maka waktu yang dibutuhkan rangkaian flip - flop dari posisi high ke low lebih panjang dan  $V_o$  akan positif. Begitu juga sebaliknya apabila phasa  $V_b$  tertinggal oleh  $V_a$  maka  $V_o$  akan negatif sedangkan waktu yang dibutuhkan rangkaian flip - flop dari posisi high ke low adalah lebih cepat. Sedangkan harga setting untuk frekwensinya diset oleh pemakaian diode zenernya. Penggunaan  $R = 330k$  dan  $C = 1\mu F$  hanya cocok untuk frekwensi rendah.

### III.10 RANGKAIAN DRIVER MOTOR STEPPER

Rangkaian driver motor stepper menggunakan IC Transistor 2N3055 sebagai IC regulator dimana rangkaian ini memberikan masukan berupa putar kanan dan putar kiri yang diaktur oleh sofware. Jika Diberikan input "0" yang dikirim oleh port 8255 akan memberikan arus ke kumparan sehingga akan memutar ke kanan motor sedangkan bila logic "1" maka akan memutar ke kiri. Pada rangkaian ini menggunakan inverter 7406 karena itu menggunakan logika negatif.





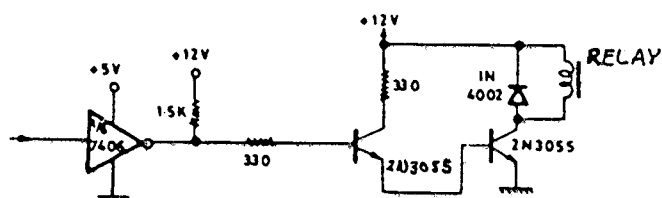
Gambar 3.7

Rangkaian Driver Motor Stepper

### III.11 RANGKAIAN PENGATUR RELAY

Rangkaian pengatur relay menggunakan transistor 2N3055 yang berfungsi sebagai switch. Yang mana sinyal inputnya dikontrol oleh PPI 8255 melalui IC 7406 (Open Colektor inverter Driver) dengan menambahkan tahanan pull Up 1.5 K ohm pada outputnya. Dalam perencanaan ini dipakai dua

buah relay 12 V dan dua buah transistor 2N3055 yang dirangkai secara hubungan Darlington agar mampu mengaktifkan relay yang membutuhkan arus cukup besar. Untuk melindungi transistor dari arus balik kumparan maka dipasang dioda pengaman.



Gambar 3.8

Rangkaian pengatur relay



### III.8. PERENCANAAN PERANGKAT LUNAK

#### III.8.1. PENDAHULUAN

Perencanaan perangkat lunak menggunakan Turbo Pascal versi 5.5 produksi Borland International.

Turbo Pascal memiliki fasilitas pengaksesan langsung ke memory dan port - port I/O. Dengan demikian sangat menunjang peralatan yang dibuat dalam Tugas Akhir ini. Pengaksesan port I/O menggunakan Port atau Portw.

Perangkat lunak synchronisasi generator ini dirancang dalam dua macam yaitu manual dan otomatis. Dimana jika manual maka proses masih dijalankan secara manual dengan melihat pada monitor channel mana yang akan di deteksi dan motor mana yang akan diputar. Sedangkan jika menggunakan otomatis maka kedua channel akan saling dibandingkan dan hasilnya akan ditampilkan di monitor bila ada yang tidak sama maka secara otomatis akan memutar motor untuk menyamakan.

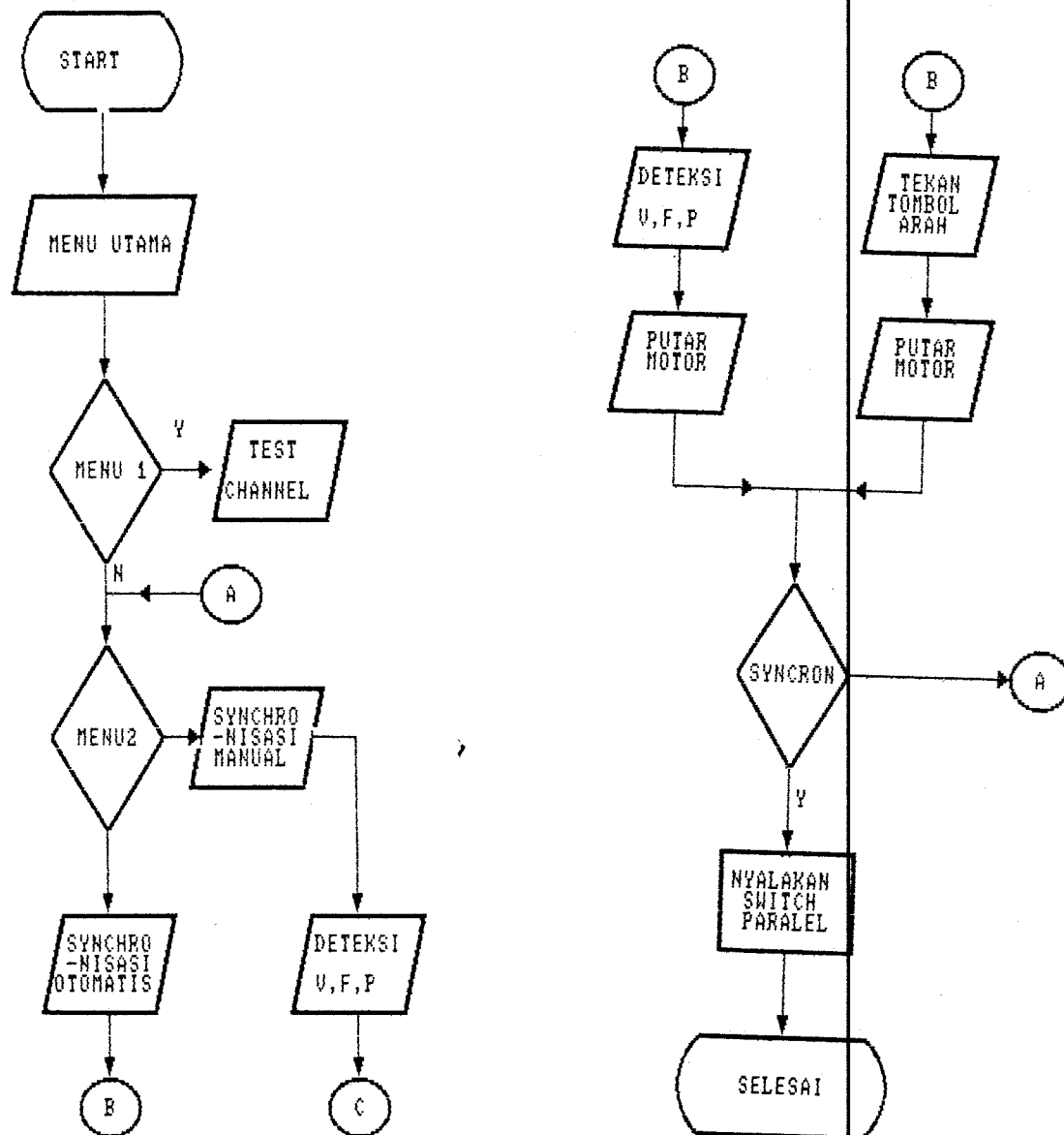
Pembahasan mengenai perangkat lunak akan dibahas sebagai berikut :

Menu Utama yang terdiri dari :

- Deteksi Tegangan
- Deteksi Frekwensi
- Sincronisasi manual
- Sincronisasi Otomatis

### 3.8.2 MENU UTAMA

Menu utama terdiri dari 2 ( dua ) sub menu yaitu menu membaca tegangan dan frekwensi dan menu Sincronisasi manual maupun otomatis adalah sebagai berikut :



Gambar 3.9

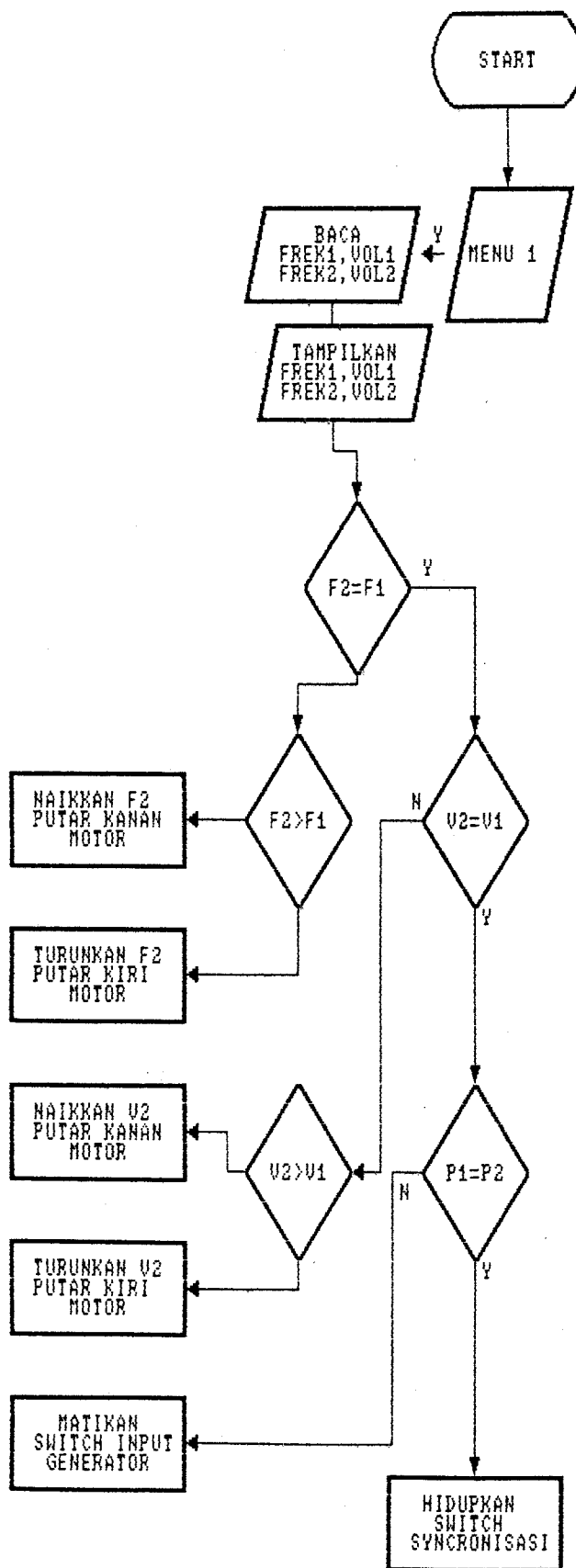
Diagram Alir menu Utama

### III. 2.3 MENU SYNCRONISASI OTOMATIS

Menu deteksi tegangan adalah menu yang merupakan bagian dari menu utama dimana menu ini akan mendeteksi tegangan yang terbaca oleh port 8255, dimana proses pembacaannya adalah dengan membaca dari rangkaian analog channel 1 kemudian membandingkan dengan output channel 2. Jika kedua Output sudah dibandingkan maka keadaan terakhir akan dikunci supaya tetap sama antara channel 1 dan channel 2, begitu juga untuk menu mendeteksi frekwensi kedua channel juga saling dibandingkan bila terjadi ketidak samaan keduanya maka akan menghubungi menu pengatur motor untuk mengatur motor baik dipercepat atau diperlambat.

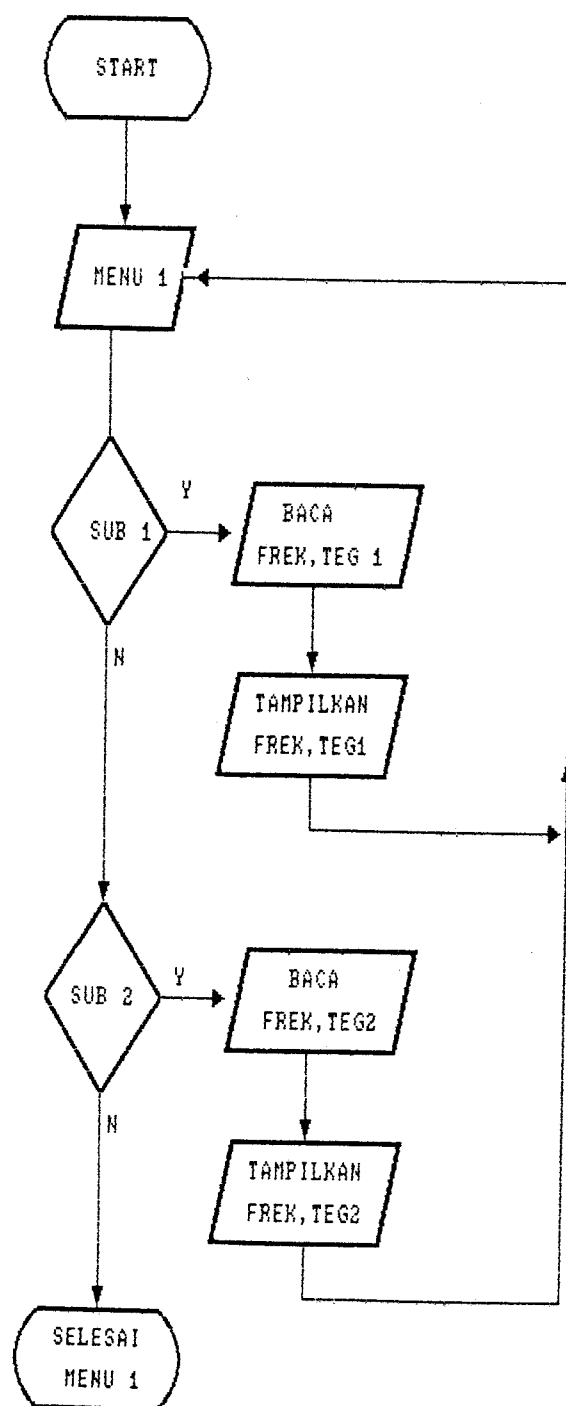
### III.2.4 MENU PENGATUR PUTARAN MOTOR

Menu pengaturan putaran motor adalah bagian dari menu utama dimana menu ini akan aktif atau dijalankan jika terjadi permintaan dari menu deteksi tegangan dan frekwensi atau terjadi kasus ketidak seimbangan kedua parameter tersebut dalam proses perbandingannya. Untuk proses manual maka program mengatur dan menyesuaikan secara bergantian.



Gambar 3.10

Diagram Alir Sincronisasi Otomatis



Gambar 3.11

Diagram alir Deteksi tegangan dan frekwensi.

## BAB IV

### PENGUJIAN DAN PENGUKURAN

#### IV.1. PENGUJIAN

Pengujian peralatan ini menggunakan 2 buah sinyal generator tipe 3311 Hewlett Pakard sebagai masukan analog yang diumpamakan sebagai input besaran dari generator, salah satu digunakan sebagai input referensi, dan lainnya digunakan sebagai variabel input, satu unit computer IBM PC - XT 640 Kbyte pada clock 4.7 Mhz. Maka ditampilkan hasil ADC

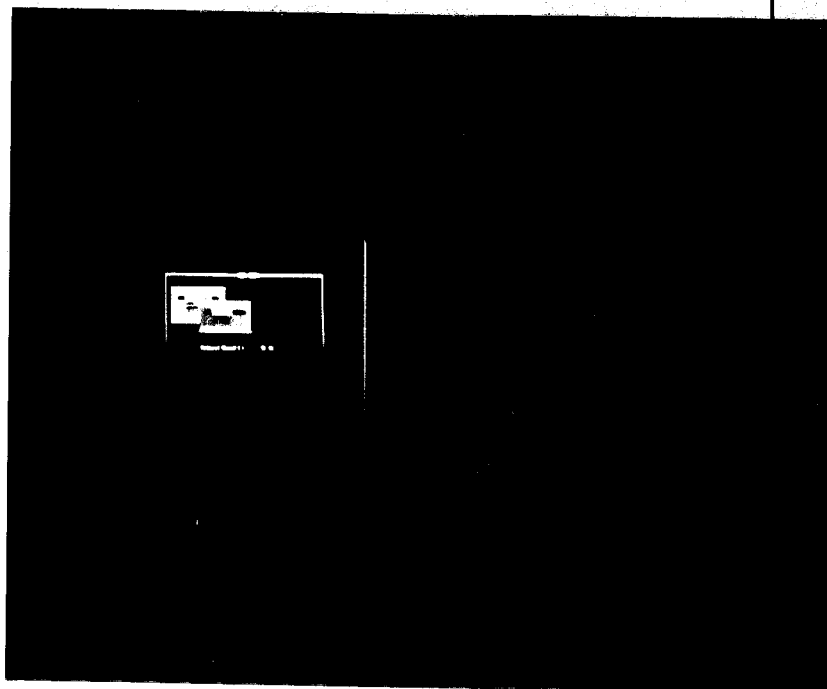
Tegangan input analog (volt)	OUTPUT DARI ADC ( VOLT )	
2.32	2.36	
2.51	2.53	
2.75	2.81	
3.16	3.23	
3.25	3.30	
3.5	3.60	

Tabel 4.1

Tabel output tegangan

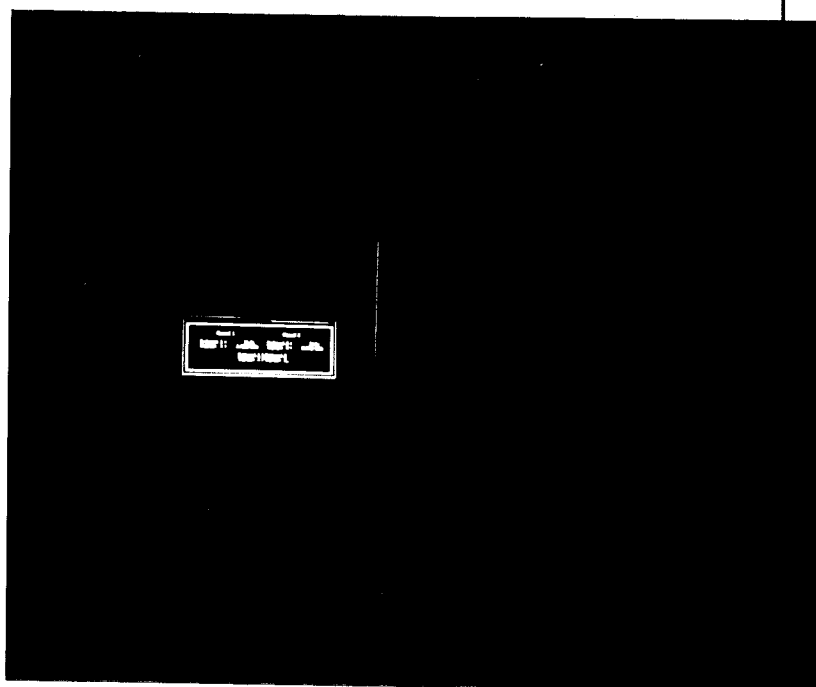
Pengukuran ADC dilakukan dengan me - set sinyal generator pada frekwensi tertentu kemudian dibandingkan dengan frekwensi yang terdeteksi dilayar, seting diubah - ubah dengan memutar tombol dan hasil seting dibandingkan dengan yang tertulis di monitor.

Frekwensi input ( Hz )	Frekwensi Output dari ADC ( Hz )
42	43
45	45
48	49
50	51
51	51
52	52



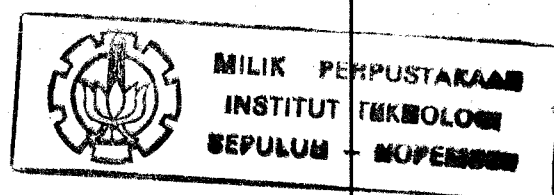
Gambar 4.1

Foto alat yang dibuat ( Software deteksi )



Gambar 4.2

Foto Alat yang dibuat ( Software Sincronisasi )





## BAB V

### KESIMPULAN

Dari pembahasan, perancangan dan pembuatan serta pengukuran yang dilakukan, diperoleh kesimpulan sebagai berikut :

Pemakaian teknik Interfacing untuk pengaturan dapat memberikan penampilan yang baik bagi pengaturan peralatan yaitu pada kepresisian, lebih cepat dan efisien dalam penggunaan waktu. Hal tersebut bisa dilihat dengan waktu yang bersamaan dapat mendeteksi dan mengatur.

Perencanaan sinkronisasi generator tiga phasa yang di interfacekan ke IBM PC - XT diharapkan dapat dipakai untuk industri maupun sebagai peralatan simulasi untuk praktikum mata kuliah elka daya dan mata kuliah lain yang berkompeten.

Dengan mempelajari masukan yang ada maka alat tersebut dapat dipakai untuk skala yang lebih besar, misalnya untuk peralatan pembangkit pada pusat pembangkit listrik, dengan menambahkan penguat, tranduser dan penurun tegangan yang lebih presisi sehingga didapatkan hasil yang optimum.

Pada hasil pengukuran menunjukkan harga yang berbeda dengan sebenarnya, hal ini disebabkan adanya pemakaian beberapa komponen yang tidak ideal.

## DAFTAR PUSTAKA

- Busono, Meningkatkan Dayaguna Komputer dengan Turbo Pascal, PT. Elex Media Computindo, Jakarta, 1989.
- Brenner C. Robert IBM PC Trobleshooting & Repair Guide  
Slawson Communication Inc, 1986.
- Driscoll, Frederick F. and Coughlin, Robert F., alih bahasa  
Ir. Herman Widodo Sumitro, Penguat Operasi dan Rangkaian Terpadu Linear, Penerbit Erlangga Jakarta, 1985.
- Eggebrechth, Lewis C., Interfacing to IBM Personal Computer,  
Howard W. Sams & Co., Inc., Indiana Polis 1983
- Gopal M., Digital Control Engineering John Wiley and Sons  
(SEA) Pte. Ltd. - Singapore, 1988
- Gayakwad, Ramakant dan Leonard Sokolov, Analog to Digital Control System, Prentice Hall International, New Jersey, 1988.
- Hall Douglas V. Microprocessors and interfacing Programming and Hardware, Mc Graw - Hill Book Company, Singapore 1986
- HM. Jogiyanto, Teori dan Aplikasi Program Komputer Bahasa Pascal, Andi Offset, Yogyakarta 1984.
- HM. Jogiyanto, Pascal Tingkat Lanjutan, Andi Offset, Yogyakarta, 1989.
- Stout, David F. and Kaufman, Milton, Handbook of Operational

Amplifier Circuit Design, Mc. Graw - Hill Inc., New York, 1976.

Singh, Avtar dan Triebel, Walter A., The 8088 Microprocessor Programming

Theraja, B.L., Handbook of Electrical Technology, Nirdja Construction & Development Co, Ltd., New Delhi, 1984

....., Conference Paper : Control of Electric Generating System, Basler Electric, Highland - Illinois USA, 1989.

....., Linear Data Book, National Semiconductor Corporation 1988.

....., Microprocessor and Peripheral Handbook volume 1, Intel Microcomputer Company, 1988.

....., TTL Data Book , Fairchild Camera & Instrument Corporation, California, 1977.



MILIK PERPUSTAKAAN  
INSTITUT TEKNOLOGI  
SEPULUH - NOPEMBER

```

Program Synchronisasi;
Uses crt, windows, dos;
const
PA1 = $200;
PB1 = $201;
PC1 = $202;
CW1 = $203;

Var Frek1, Frek2    : Integer;
    Volt1, Volt2    : Real;
    EOC, I, Pilihan : integer;
    ch : char;

    PosisiF, PosisiT      : Integer;
    Lebar_Atas, Lebar_Bawah : LongInt;
    DataFrek : array[0..3] of integer;
    DataTeg  : array[0..3] of integer;

Procedure color(t,b:byte);
begin
    textcolor(t);
    textbackground(b);
end;

Procedure MenuUtama;
Begin
    color(white, red);
    GotoXY(7,6);Write('Test Channel');color(black, cyan);
    GotoXY( 7,7);Write('    Sync. Manual    ');
    GotoXY( 7,8);Write('    Synchronisasi    ');
    GotoXY( 7,9);Write('    S E L E S A I    ');
end;

Procedure Pull_Menu(Pilihan : integer);
Begin
    case Pilihan of
        1 : begin GotoXY( 7,6);Write('    Test Channel    ');end;
        2 : begin GotoXY( 7,7);Write('    Sync. Manual    ');end;
        3 : begin GotoXY( 7,8);Write('    Synchronisasi    ');end;
        4 : begin GotoXY( 7,9);Write('    S E L E S A I    ');end;
    end;
end;

procedure Menu_Up(var Pil : integer; Bawah, Atas : integer);
begin
    color(Black, cyan);
    Pull_Menu(Pil);
    if Pil = Bawah Then
        Pil:=atas
    else

```

```

    Pil:=Pil-1;
    color(white,LightRed);
    Pull_Menu(Pil);
end;
procedure Menu_Down(var Pil :integer; Bawah,Atas : integer);
begin
    color(Black,cyan);
    Pull_Menu(Pil);
    if Pil = Atas Then
        Pil:=bawah
    else
        Pil:=Pil+1;
    color(white,LightRed);
    Pull_Menu(Pil);
end;

```

```

Procedure Baca_Frek1(Var Frekwensi : Integer);
Var
    DetikS,Detik      : LongInt;
    Jam,men,Det,SDet  : Word;
    BacaFrek,Data     : Integer;

begin
    Frekwensi := 0;
    repeat
        BacaFrek := Port[PC1];
        Data := BacaFrek AND $10;
        Until Data = $10;

        GetTime(Jam,Men,Det,SDet);
        DetikS := Men*60*100 + Det*100 + SDet;
        Repeat
            Repeat
                BacaFrek := Port[PC1];
                Data := BacaFrek AND $10;
                delay(5);
            Until Data = $10;
            Repeat
                BacaFrek := Port[PC1];
                Data := BacaFrek AND $10;
                delay(5);
            Until Data = 0;
            inc(Frekwensi);
            GetTime(Jam,Men,Det,SDet);
            Detik := Men*60*100 + Det*100 + SDet - DetikS;
            Until Detik >= 100;
        end;
    end;

```

```

Procedure Baca_Frek2(Var Frekwensi : Integer);
Var

```

```

DetikS, Detik      : LongInt;
Jam, men, Det, SDet : Word;
BacaFrek, Data     : Integer;

```

```

begin
  Frekwensi := 0;
  repeat
    BacaFrek := Port[PC1];
    Data := BacaFrek AND $20;
    Until Data = $20;
    GetTime(Jam, Men, Det, SDet);
    DetikS := Men*60*100 + Det*100 + SDet;
    Repeat
      Repeat
        BacaFrek := Port[PC1];
        Data := BacaFrek AND $20;
        delay(5);
        Until Data = $20;
      Repeat
        BacaFrek := Port[PC1];
        Data := BacaFrek AND $20;
        delay(5);
        Until Data = $0;
        inc(Frekwensi);
        GetTime(Jam, Men, Det, SDet);
        Detik := Men*60*100 + Det*100 + SDet - DetikS;
        Until Detik >= 100;
    end;
end;

```

```

Procedure ADC;

```

```

begin
  Port[PB1] := $91; {10010001b}
  Port[PB1] := $99; {10011001b}
  Port[PB1] := $80; {10000000b}

  repeat
    EOC := Port[PC1];
    EOC := EOC And 0;
    until EOC = 0;

  repeat
    EOC := Port[PC1];
    EOC := EOC And 1;
    until EOC = 1;
  end;
end;

```

```

Procedure Baca_Teg1(Var Hasil : Real);

```

```

begin
  Port[PB1] := $80;
  ADC;
  Hasil := 0;
  Hasil := Port[PA1];

  Hasil := Hasil / 51;
end;

```

```

Procedure Baca_Teg2(Var Hasil : Real);
begin
    Port[PB1] := $81; {100000001b}
    ADC;
    Hasil := 0;
    Hasil := Port[PA1];
    Hasil := Hasil / 51;
end;

procedure Inisial;
begin
    DataFrek[0] := $f7;
    DataFrek[1] := $fb;
    DataFrek[2] := $fd;
    DataFrek[3] := $fe;
    DataTeg[0] := $bf;
    DataTeg[1] := $7f;
    DataTeg[2] := $df;
    DataTeg[3] := $ef;
end;

procedure Tambah_Frek(Step : Integer);
var i : integer;
begin
    For i := 1 To Step do
    begin
        PosisiF := PosisiF + 1;
        if PosisiF > 3 Then PosisiF := 0;
        Port[$300] := DataFrek[PosisiF];
        Delay(25);
    end;
end;

procedure Kurangi_Frek(Step : Integer);
var i : integer;
begin
    For i := 1 To Step do
    begin
        PosisiF := PosisiF - 1;
        if PosisiF < 0 Then PosisiF := 3;
        Port[$300] := DataFrek[PosisiF];
        Delay(25);
    end;
end;

procedure Tambah_Teg(Step : Integer);
var i : integer;
begin
    For i := 1 To Step do
    begin
        PosisiT := PosisiT + 1;

        if PosisiT > 3 Then PosisiT := 0;
    end;
end;

```

```

    Port[$302]:=DataTeg[PosisiT];
    Delay(25);
end;
end;

```

```

procedure Kurangi_Teg(Step : Integer);
var i : integer;
begin
    For i:= 1 To Step do
    begin
        PosisiT:=PosisiT-1;
        if PosisiT < 0 Then PosisiT:=3;
        Port[$302]:=DataTeg[PosisiT];
        Delay(25);
    end;
end;

```

```

procedure Baca_Phase(Var Lebar_atas, Lebar_Bawah : LongInt);
Var
    Lebar : LongInt;
begin
    Lebar_Atas:=0;
    Lebar_Bawah:=0;
    repeat
        Lebar:=Port[$202] and $80;
    until Lebar = $80;

    repeat
        Lebar:=Port[$202] and $80;
        Lebar_Atas:=Lebar_Atas+1;
    until Lebar = $0;

    repeat
        Lebar:=Port[$202] and $80;
        Lebar_Bawah:=Lebar_Bawah+1;
    until Lebar = $80;
end;

```

```

Procedure Test_Channel;
Var
    Pilihan : integer;
    ch : char;

```

```

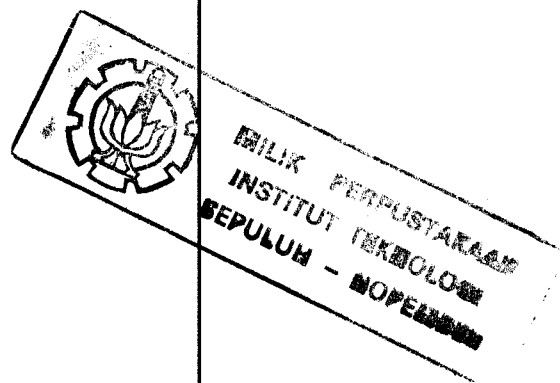
Procedure MenuTest;
Begin
    color(white,red);
    GotoXY(20,9);Write('Channel 1'); color(black,Magenta);
    GotoXY( 20,10);Write('    Channel 2    ');
    GotoXY( 20,11);Write('    E X I T    ');
end;

```

```

Procedure Pull_Menu(Pilihan : integer);
Begin

```





```

case Pilihan of
1  : begin GotoXY( 20,9);Write(' Cahnnel 1 ');end;
2  : begin GotoXY( 20,10);Write(' Channel 2 ');end;
3  : begin GotoXY( 20,11);Write(' E X I T ');end;
end;
end;

```

```

procedure Menu_Up(var Pil : integer; Bawah,Atas : integer);
begin
  color(Black,Magenta);
  Pull_Menu(Pil);
  if Pil = Bawah Then
    Pil:=atas
  else
    Pil:=Pil-1;
  color(white,LightRed);
  Pull_Menu(Pil);
end;

```

```

procedure Menu_Down(var Pil : integer; Bawah,Atas :
integer);
begin
  color(Black,Magenta);
  Pull_Menu(Pil);
  if Pil = Atas Then
    Pil:=bawah
  else
    Pil:=Pil+1;
  color(white,LightRed);
  Pull_Menu(Pil);
end;

```

```

BEGIN
  Pilihan:=1;
  win(17,7,42,13,1,7,5);
  MenuTest;
  Repeat
    gotoxy(39,Pilihan+8);
    ch:=readkey;
    if ch = #0 then
      ch:=readkey;
    case ch of
      #72 : Menu_Up(Pilihan,1,3);
      #80 : Menu_Down(Pilihan,1,3);
      #13 : begin
        case Pilihan of
          1 : begin
            win(15,15,58,20,6,1,1);
            Color(White,blue);
            Baca_Frek1(Frek1);

            Baca_Teg1(Volt1);
            GotoXY(18,17);Write(' Frekwensi Channel 1
              = ',Frek1:10,' Hz');

```

```

        GotoXY(18,18);Write('Tegangan Channel 1
= ',Volt1:10:4,' Volt');
        repeat until readkey in [#27,#13];
        Wind:=clos_wind;
        end;
        2 : begin
            win(15,15,58,20,6,1,1);
            Color(White,blue);
            Baca_Frek2(Frek2);
            Baca_Teg2(Volt2);
            GotoXY(18,17);Write('Frekwensi Channel 2
= ',Frek2:10,' Hz');
            GotoXY(18,18);Write('Tegangan Channel 2
= ',Volt2:10:4,' Volt');
            repeat until readkey in [#27,#13];
            Wind:=clos_wind;
            end;
        3 : begin ch:=#27 end;
        end;
    end;
    until (ch=#27);
    wind:=clos_wind;
end;

```

```

Procedure Run_Manual;
var ch : char;

```

```

begin
    win(45,4,75,12,7,1,1);
    color(2,blue);
    GotoXY(49,6);Write(chr(27),' : Kurangi Frekwensi ');
    GotoXY(49,7);Write(chr(26),' : Tambah Frekwensi');
    GotoXY(49,8);Write(chr(24),' : Kurangi Tegangan');
    GotoXY(49,9);Write(chr(25),' : Tambah Tegangan');
    GotoXY(49,11);Write('Ins : Swich Phase');
    win(3,13,77,24,15,1,1);
    GotoXY(18,15);Write('Channel 1');
    GotoXY(52,15);Write('Channel 2');

    Baca_Frek1(Frek1);
    Baca_Frek2(Frek2);
    Baca_Teg1(Volt1);
    Baca_Teg2(Volt2);
    GotoXY(10,17);Write('Frekwensi 1 = ',Frek1:10,' Hz');
    GotoXY(10,18);Write('Tegangan 1 = ',Volt1:10:4,' Volt');
    GotoXY(44,17);Write('Frekwensi 2 = ',Frek2:10,' Hz');
    GotoXY(44,18);Write('Tegangan 2 = ',Volt2:10:4,' Volt');

```

```

if (Frek1 = Frek2) Then
    begin

```

```

    GotoXY(29,20);
    Write('Frekwensi 1 = Frekwensi 2');
end
else
    if Frek1 > Frek2 Then
        begin
            GotoXY(29,20);Write('Frekwensi 1 > Frekwensi 2');
        end
    else
        begin
            GotoXY(29,20);Write('Frekwensi 1 < Frekwensi 2');
        end;
    end;

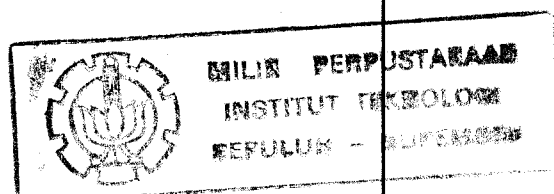
if Abs(Volt1 - Volt2) < 0.0001 Then
    begin
        GotoXY(29,21);
        Write('Tegangan 1 = Tegangan 2');
    end
else
    if Volt1 > Volt2 Then
        begin
            GotoXY(29,21);Write('Tegangan 1 > Tegangan 2');
        end
    else
        begin
            GotoXY(29,21);Write('Tegangan 1 < Tegangan 2');
        end;
    end;

Baca_Phase(Lebar_Atas,Lebar_Bawah);
if Abs(Lebar_Atas - Lebar_Bawah) in [0..15] Then
    begin
        GotoXY(29,22);
        Write('Phase 1 = Phase 2');
    end
else
    if Lebar_Atas > Lebar_Bawah Then
        begin
            GotoXY(29,22);Write('Phase 1 > Phase 2');
        end
    else
        begin
            GotoXY(29,22);Write('Phase 1 < Phase 2');
        end;
    end;

if (Frek1 = Frek2) and (Abs(Volt1 - Volt2) < 0.0001) and
(Abs(Lebar_Atas - Lebar_Bawah) in [0..15]) Then
    begin
        sound(200);
        delay(500);
        nosound;
    end;

repeat
    if Keypressed Then
        begin

```



```

ch:=readkey;
if ch = #0 Then
  ch:=readkey;
case ch of
#75 : begin
  Kurangi_Frek(2);
  Baca_Frek2(Frek2);
  GotoXY(44,17);Write('Frekwensi
',Frek2:10,' Hz');
  end;
#77 : begin
  Tambah_Frek(2);
  Baca_Frek2(Frek2);
  GotoXY(44,17);Write('Frekwensi
',Frek2:10,' Hz');
  end;
#72 : begin
  Kurangi_Teg(2);
  Baca_Teg2(Volt2);
  GotoXY(44,18);Write('Tegangan
',Volt2:10:4,' Volt');
  end;
#80 : begin
  Tambah_Teg(2);
  Baca_Teg2(Volt2);
  GotoXY(44,18);Write('Tegangan
',Volt2:10:4,' Volt');
  end;
#82 : begin
  Port[$301]:=$55;delay(20);
  Port[$301]:=$0;
  delay(100);
  Baca_Phase(Lebar_Atas,Lebar_Bawah);
  end;
end;

if (Frek1 = Frek2) Then
  begin
    GotoXY(29,20);
    Write('Frekwensi 1 = Frekwensi 2');
  end
else
  if Frek1 > Frek2 Then
    begin
      GotoXY(29,20);Write('Frekwensi 1>Frekwensi 2');
    end
  else
    begin
      GotoXY(29,20);Write('Frekwensi 1 < Frekwensi 2');
    end;
  end;

if Abs(Volt1 - Volt2) < 0.0001 Then
  begin
    GotoXY(29,21);

```

```

        Write('Tegangan 1 = Tegangan 2');
    end
else
    if Volt1 > Volt2 Then
        begin
            GotoXY(29,21);Write('Tegangan 1> Tegangan 2');
        end
    else
        begin
            GotoXY(29,21);Write('Tegangan 1 < Tegangan 2');
        end;

        if Abs(Lebar_Atas - Lebar_Bawah) in [0..15] Then
            begin
                GotoXY(29,22);
                Write('Phase 1 = Phase 2');
            end
        else
            if Lebar_Atas > Lebar_Bawah Then
                begin
                    GotoXY(29,22);Write('Phase 1> Phase 2');
                end
            else
                begin
                    GotoXY(29,22);Write('Phase 1 < Phase 2');
                end;

                if (Frek1 = Frek2) and (Abs(Volt1 - Volt2) < 0.0001)
                and (Abs(Lebar_Atas - Lebar_Bawah) in [0..15]) Then
                    begin
                        Port[$301]:=$AA;
                        delay(20);
                        Port[$301]:=$0;
                        sound(200);
                        delay(500);
                        nosound;
                    end;
                end;
            Until ch in [#27];
            wind:=clos_wind;
            wind:=clos_wind;
        end;

```

Procedure Run\_Program;

```

begin
    win(3,13,77,24,15,1,1);
    GotoXY(18,15);Write('Channel 1');
    GotoXY(52,15);Write('Channel 2');

    Baca_Frek1(Frek1);
    Baca_Frek2(Frek2);
    Baca_Teg1(Volt1);

```

```

Baca_Teg2(Volt2);
GotoXY(10,17);Write('Frekwensi 1 = ',Frek1:10,' Hz');

GotoXY(10,18);Write('Tegangan 1 = ',Volt1:10:4,' Volt');
GotoXY(44,17);Write('Frekwensi 2 = ',Frek2:10,' Hz');
GotoXY(44,18);Write('Tegangan 2 = ',Volt2:10:4,' Volt');

repeat
  if (Frek1 = Frek2) Then
    begin
      GotoXY(29,20);
      Write('Frekwensi 1 = Frekwensi 2');
    end
  else
    if Frek1 > Frek2 Then
      begin
        GotoXY(29,20);Write('Frekwensi 1 > Frekwensi 2');
        Tambah_Frek(2);
        Baca_Frek2(Frek2);
        GotoXY(44,17);Write('Frekwensi 2 = ',Frek2:10,' Hz');
      end
    else
      begin
        GotoXY(29,20);Write('Frekwensi 1 < Frekwensi 2');
        Kurangi_Frek(2);
        Baca_Frek2(Frek2);
        GotoXY(44,17);Write('Frekwensi 2 = ',Frek2:10,' Hz');
      end;
  Until Frek1 = Frek2;
  GotoXY(29,20);
  Write('Frekwensi 1 = Frekwensi 2');

Repeat
  if Abs(Volt1 - Volt2) < 0.001 Then
    begin
      GotoXY(29,21);
      Write('Tegangan 1 = Tegangan 2');
    end
  else
    if Volt1 > Volt2 Then
      begin
        GotoXY(29,21);Write('Tegangan 1 > Tegangan 2');
        Tambah_Teg(2);
        Baca_Teg2(Volt2);
        GotoXY(44,18);Write('Tegangan 2 = ',Volt2:10:4,'
Volt');
      end
    else
      begin
        GotoXY(29,21);Write('Tegangan 1 < Tegangan 2');
        Kurangi_Teg(2);
        Baca_Teg2(Volt2);
        GotoXY(44,18);Write('Tegangan 2 = ',Volt2:10:4,'
Volt');
      end

```

```

    end;
Until Abs(Volt1 - Volt2) < 0.001;
GotoXY(29,21);
Write('Tegangan 1 = Tegangan 2');

Repeat
  Baca_Phase(Lebar_Atas,Lebar_Bawah);
  if Abs(Lebar_Atas - Lebar_Bawah) < 15 Then
    begin
      GotoXY(29,22);
      Write('Phase 1 = Phase 2');
    end
  else
    if Lebar_Atas > Lebar_Bawah Then
      begin
        GotoXY(29,22);Write('Phase 1 > Phase 2');
        Port[$301]:=$55;delay(20);
        Port[$301]:=$0;
        delay(500);
      end
    else
      begin
        GotoXY(29,22);Write('Phase 1 < Phase 2');
        Port[$301]:=$55;delay(20);
        Port[$301]:=$0;
        delay(500);
      end;
    Until Abs(Lebar_Atas - Lebar_Bawah) < 15;
    GotoXY(29,22);
    Write('Phase 1 = Phase 2');

    if (Frek1 = Frek2) and (Abs(Volt1 - Volt2) < 0.001) and
    (Abs(Lebar_Atas - Lebar_Bawah) < 15) Then
      begin
        Port[$301]:=$AA;
        delay(20);
        Port[$301]:=$0;
        sound(200);
        delay(500);
        nosound;
      end;

    color(LightRed+blink,blue);
    GotoXY(34,15);Write('Tekan key');
    repeat until readkey in [#27,#13];
    color(White,blue);
    GotoXY(34,15);Write(' ');
    wind:=clos_wind;
  end;

BEGIN
  clrscr;
  init;
  Port[$203]:=$99;

```

```

Port[$303]:=$80;
inisial;
PosisiF:=0;
PosisiT:=0;

win(1,1,79,25,7,5,1);
color(15,2);
gotoxy(36,1);write(' UJI COBA ');
Pilihan:=1;
win(4,4,29,11,5,15,3);
MenuUtama;
ch:='';
repeat
  gotoxy(26,Pilihan+5);
  ch:=readkey;
  if ch = #0 then
    ch:=readkey;

  case ch of
    't','T' : Test_Channel;
    #72 : Menu_Up(Pilihan,1,4);
    #80 : Menu_Down(Pilihan,1,4);
    #13 : begin
      case Pilihan of
        1 : begin
          Test_Channel;
          end;
        2 : begin
          Run_Manual;
          end;
        3 : begin
          Run_Program;
          end;
        4 : begin ch:=#27 end;
          end;
      end;
    end;

until (ch=#27);
wind:=clos_wind;
wind:=clos_wind;
END.

```



## USULAN TUGAS AKHIR

1. JUDUL TUGAS AKHIR : PERENCANAAN DAN PEMBUATAN SINKRONISASI  
GENERATOR TIGA PHASA YANG  
DIINTERFACEKAN KE IBM PC - XT.
2. BIDANG STUDI : ELEKTRONIKA
3. RUANG LINGKUP : ELEKTRONIKA DIGITAL  
RANGKAIAN LINIER AKTIF  
ELEKTRONIKA INDUSTRI  
PEMROGRAMAN KOMPUTER  
ELEKTRONIKA DAYA
4. LATAR BELAKANG : Perkembangan teknologi elektronika  
pada saat ini amat pesat terutama di  
bidang mikroelektronika. Kehadiran  
mikrokomputer sebagai peralatan  
pembantu dalam industri dirasakan  
penting sekali mengingat terbatasnya  
kemampuan peralatan manual. Salah  
satu aplikasi dalam bidang industri  
adalah pengaturan sinkronisasi  
generator tiga phasa yang biasanya  
dilakukan secara manual dapat  
dilakukan dengan bantuan komputer,  
dengan membuat suatu peralatan yang  
diinterfacekan ke IBM PC - XT,  
peralatan tersebut akan menerima input  
dari generator yang berupa tegangan.

frekwensi, phasa dari generator. Apabila input - input tersebut memenuhi syarat - syarat kerja paralel maka peralatan tersebut akan menggerakkan relay - relay yang akan membuat switch terhubung dan generator akan beroperasi paralel. Dengan dipakainya peralatan komputer maka dapat ditingkatkan keandalan dari sistem, baik andal dalam kepresisiannya, kecepatan proses maupun kemudahan pengendalian sistem sehingga didapatkan performance yang lebih baik.

5. PENELAAHAN STUDI : Dalam melakukan kerja paralel, generator memerlukan syarat - syarat yang harus dipenuhi sebelumnya. Syarat - syarat tersebut di deteksi kemudian dibandingkan antar generator satu dengan yang lain oleh program dalam CPU, kemudian input tersebut akan mengaktifkan switch jika syarat - syarat yang diperlukan terpenuhi.

6. TUJUAN : - Menerapkan teori dan pengertian yang berhubungan dengan elektronika digital, elektronika analog, elektronika industri serta pemrograman

komputer.

- Merencanakan dan membuat alat untuk sinkronisasi generator tiga phasa yang di interfacekan ke IBM PC - XT sehingga didapatkan peralatan yang handal dalam kepresisiannya, kecepatan proses dan kemudahan dalam pengedaliannya.

7. LANGKAH - LANGKAH : - Studi literatur

- Perencanaan peralatan
- Membuat peralatan
- Pengujian dan kalibrasi
- Penyusunan naskah

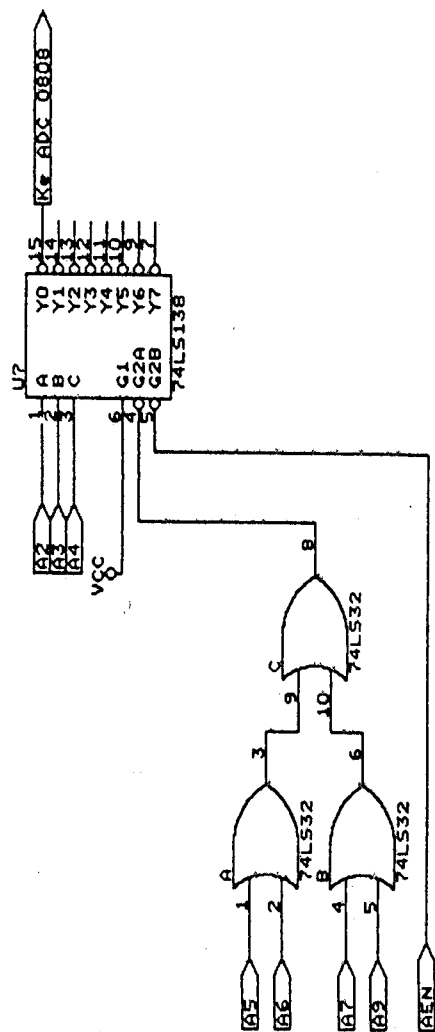
8. JADWAL KEGIATAN : Seluruh kegiatan direncanakan dapat diselesaikan dalam waktu enam bulan dengan jadwal sebagai berikut :

KEGIATAN	BULAN KE					
	1	2	3	4	5	6
Studi literatur	■■■■■■■■■■					
Perencanaan alat		■■■■■■■■■■				
Pembuatan alat			■■■■■■■■■■			
Pengujian & kalibrasi					■■■■■■■■■■	
Penyusunan naskah					■■■■■■■■■■	

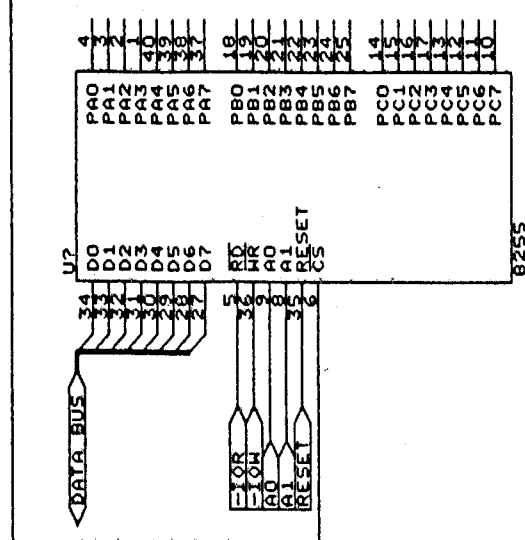
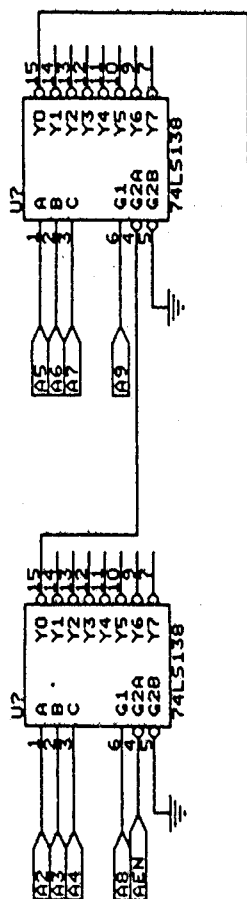
## 9. RELEVANSI

: Hasil dari tugas akhir ini diharapkan dapat digunakan oleh pemakai generator dimana dapat diaplikasikan sistem pengaturannya. Aplikasi mikrokomputer dalam bidang industri terutama dalam bidang tenaga listrik akan meningkatkan keandalan yaitu andal dalam kepresisiannya, kecepatan proses dan kemudahan dalam pengendalian sistem.

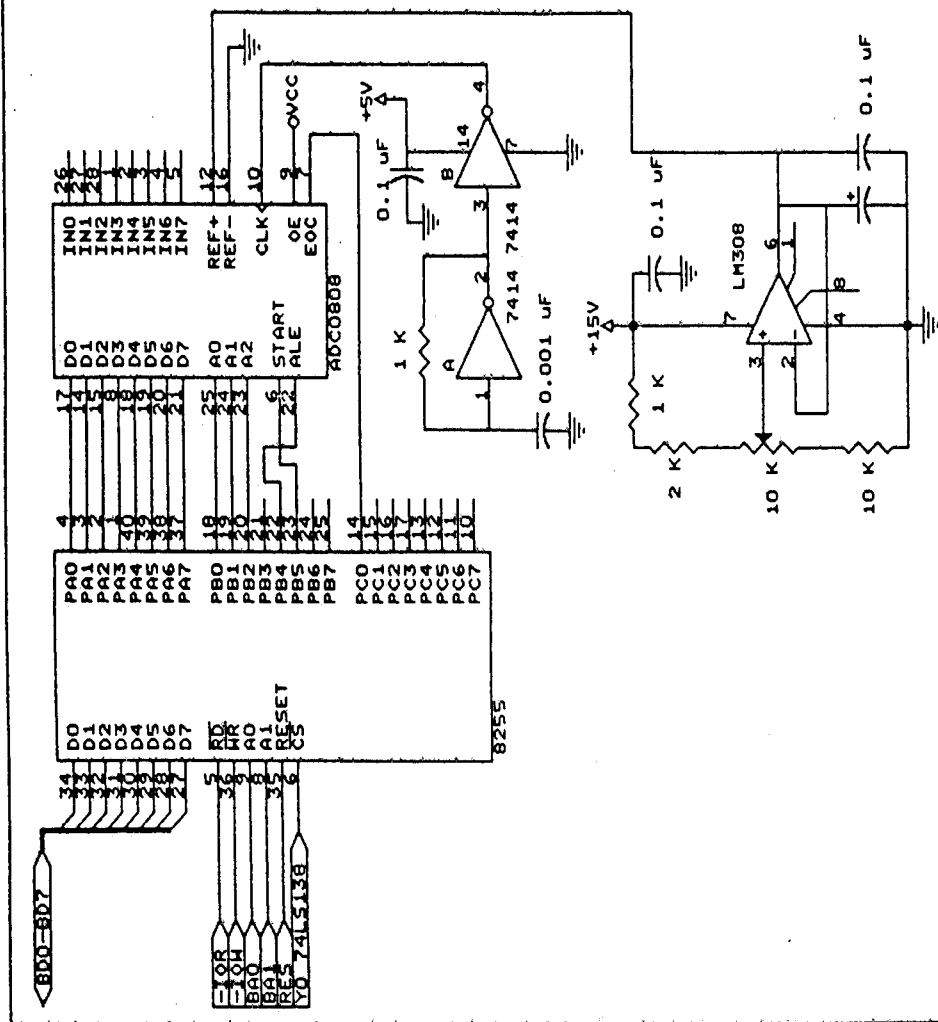




Title	DECODER
Size Document Number	REV
A	
Date:	January 1, 1980 Sheet of



Title	DECODER + PPI8255
Size	A
Document Number	REV
Date	January 1, 1980
Sheet	of



Title	ADC 0808
Size	Document Number
REV	1
Date:	January 1, 1980
Sheet	of